



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

POLOHOVÝ SNÍMAČ PRO DYNAMICKOU VERTIKALIZACI

POSITION SENSOR FOR DYNAMIC VERTICALIAZTION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Michaela Hláčiková

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Oto Janoušek, Ph.D.

BRNO 2018



Bakalářská práce

bakalářský studijní obor **Biomedicínská technika a bioinformatika**

Ústav biomedicínského inženýrství

Studentka: Michaela Hláčiková **ID:** 186659 **Ročník:** 3 **Akademický rok:** 2017/18

NÁZEV TÉMATU:

Polohový snímač pro dynamickou vertikalizaci

POKYNY PRO VYPRACOVÁNÍ:

1) Proveďte literární rešerši problematiky využití terapeutických her v procesu rehabilitace dynamickým vertikalizátorem. 2) Navrhněte s využitím platformy Arduino prototyp snímače polohy opěrné desky vertikalizátoru, umožňující v reálném čase bezdrátově přenášet polohová data do počítače. Umístěte jej na vertikalizátor tak, aby byly dostupné ovládací prvky. 3) Navrhněte a v Matlabu implementujte rozhraní pro komunikaci se snímačem, umožňující záznam a vykreslení okamžité polohy snímače a současné ovládání počítačové hry pomocí kurzorových šipek klávesnice, alternativně vytvořte jednoduchou terapeutickou hru v Matlabu. 4) Zaznamenejte pohyb skupiny dobrovolníků podle předem definovaného protokolu měření. 5) Vytvořte GUI, umožňující zobrazení a analýzu zaznamenaných polohových dat. 6) U vybraných parametrů statisticky srovnajte rozdíly mezi probandy stojícími na jedné noze a na obou nohách. 7) Interpretujte výsledky, definujte limity navrženého snímače a diskutujte je.

DOPORUČENÁ LITERATURA:

- [1] VODA, Zbyšek. Průvodce světem Arduina. Bučovice: Martin Stříž, 2015, 240s. ISBN 978-80-87106-90-7.
- [2] SMITH, Alan. Introduction to Arduino: A piece of cake!. CreateSpace Independent Publishing Platform, 2011, 170s. ISBN 978-1463698348.

Termín zadání: 5.2.2018

Termín odevzdání: 25.5.2018

Vedoucí práce: Ing. Oto Janoušek, Ph.D.

Konzultant:

prof. Ing. Ivo Provazník, Ph.D.

předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Dynamický vertikalizátor je prístroj určený na rehabilitáciu pacientov so zdravotným postihnutím dolných končatín. Rehabilitácia s využitím dynamického vertikalizátora prebieha prostredníctvom hrania počítačových hier, ktoré sú ovládané pohybom tela. Cieľom bakalárskej práce je skonštruovať prototyp snímača pohybu pre dynamický vertikalizátor, namerať patientske dáta počas hrania terapeutickkej hry a na základe nameraných dát vyhodnotiť parametre pohybu pacientov.

Kľúčové slová

Dynamický vertikalizátor, polohový snímač, rehabilitácia

Abstract

Dynamic verticalizer is a device designed for the rehabilitation of patients with lower limbs disabilities. Rehabilitation takes place by playing computer games that are controlled by body movements. The aim of the bachelor thesis is to construct a motion sensor prototype, to measure patient data during therapy and to appraise the extent of patient movement on the basis of measured data.

Keywords

Dynamic verticalizator, position sensor, rehabilitation

Bibliografická citace:

HLÁČIKOVÁ, M. *Polohový snímač pro dynamickou vertikalizaci*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2018. 53s.
Vedoucí práce: Ing. Oto Janoušek, Ph.D.

Prohlášení

„Prohlašuji, že svou závěrečnou práci na téma Polohový snímač pro dynamickou vertikalizaci jsem vypracovala samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autorka uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědomá následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.“

V Brně dne **22.5.2018**

.....

podpis autorky

Pod'akovanie

Ďakujem vedúcemu bakalárskej práce Ing. Otovi Janouškovi, Ph.D. za účinnú metodickú, pedagogickú, odbornú pomoc, trpezlivosť a cenné rady pri spracovaní mojej bakalárskej práce.

V Brne dňa **22.5.2018**

.....

podpis autorky

Obsah

1	Úvod.....	1
2	Terapeutické hry v procese rehabilitácie dynamickým vertikalizátorom	2
2.1	Dynamický vertikalizátor	2
2.2	Terapeutické hry.....	4
2.3	Ovládanie terapeutických hier.....	4
2.4	Hodnotenie rozsahu pohybu z nameraných dát	5
3	Prototyp snímača polohy	6
3.1	Akcelerometer	6
3.2	Gyroskop	7
4	Hardware prototypu snímača polohy	9
4.1	Arduino	9
4.1.1	Vývojové dosky Arduino	9
4.2	Arduino Leonardo Pro Micro	10
4.3	Snímač polohy.....	12
4.3.1	Digital motion procesor a zbernica I2C	13
4.4	Bezdrôtový prenos údajov do počítača	14
4.4.1	Piny	15
5	Software prototypu snímača polohy	16
5.1	Programovanie prototypu snímača polohy.....	16
5.2	Ovládanie klávesových šípok.....	17
5.2.1	Simulácia v programe Arduino IDE	17
5.2.2	Simulácia v programe Matlab R2017a	18
5.3	Hra Cube Dodger	19
5.4	Meranie polohových dát počas hrania hry	20
5.5	Meranie polohových dát určených na analýzu pohybu.....	22
6	Analýza polohových dát	23
6.1	Príprava dát	23
6.2	Spracovanie dát	24
6.2.1	Rozsah pohybu.....	24
6.2.2	Zrýchlenie a rýchlosť pohybu	30
6.2.3	Linearita pohybu	31
6.3	Grafické prostredie	33
7	Štatistické Vyhodnotenie	35
7.1	Vyhodnotenie rozsahu pohybu.....	35

7.2	Vyhodnotenie zrýchlenia a rýchlosti pohybu.....	38
7.3	Vyhodnotenie linearitu pohybu.....	40
8	Hodnotenie prototypu snímača polohy a výsledkov.....	42
8.1	Prototyp snímača polohy.....	42
8.2	Hodnotenie výsledkov.....	42
	Záver.....	43
	Literatúra.....	45
	Zoznam symbolov, veličín a skratiek	47
	Zoznam príloh.....	48
	Prílohy.....	49

Zoznam obrázkov

Obr. 2.1.1 Dynamický vertikalizátor BALANCE-Trainer z rôznych uhlov.....	2
Obr. 2.1.2 Popis prístroja BALANCE-Trainer.....	3
Obr. 2.3.1 Umiestnenie prototypu snímača polohy.....	4
Obr. 3.2.1 Znázornenie osí a ich rotácií modulu MPU-6050.....	8
Obr. 4.2.1 Arduino Leonardo Pro Micro.....	10
Obr. 4.2.2 Rozloženie pinov na doske Arduino Leonardo Pro Micro.....	11
Obr. 4.2.3 Schéma zapojenia komponentov.....	11
Obr. 4.2.4 Schéma zapojenia prototypu snímača polohy.....	12
Obr. 4.3.1 Akcelerometer + Gyroskop MPU-6050.....	13
Obr. 4.4.1 Bluetooth modul HC-05.....	14
Obr. 5.2.1 Vývojový diagram simulácie stláčania šípok.....	19
Obr. 5.3.1 Hranie hry Cube Dodger pomocou dynamického vertikalizátora.....	20
Obr. 5.4.1 Trajektória pohybu (vľavo) a vyznačený detail pohybu z jednej strany na druhú (vpravo).....	21
Obr. 6.2.1 Vykreslenie trajektórie pohybu v 3D.....	24
Obr. 6.2.2 Vykreslenie trajektórie pohybu v 2D.....	25
Obr. 6.2.3 Signál z osi X s vyznačenými maximálnymi hodnotami.....	26
Obr. 6.2.4 Signál z osi Y s vyznačenými maximálnymi hodnotami.....	27
Obr. 6.2.5 Signál z osi Z s vyznačenými maximálnymi hodnotami.....	28
Obr. 6.2.6 Rozsah pohybu probanda s vyznačenými maximami polohy.....	29
Obr. 6.2.7 Trajektória pohybu pri opisovaní štvorca pohybom tela.....	30
Obr. 6.2.8 Linearita pohybu na ose X.....	32
Obr. 6.3.1 Analýza rozsahu pohybu v grafickom prostredí.....	33
Obr. 7.1.1 Rozdelenie dát pri náklone vpravo.....	35
Obr. 7.1.2 Percentuálne dosiahnutie plného rozsahu pri náklone vľav.....	37
Obr. 7.2.1 Priemerná rýchlosť na ose X.....	38
Obr. 7.2.2 Najväčšie zrýchlenie na ose	40
Obr. 7.3.1 Percentuálne dosiahnutie linearity pohybu.....	41

Zoznam tabuliek

Tab. 4.1 Piny Bluetooth modulu HC-05.....	15
Tab. 5.1 Hodnotenie merania.....	21
Tab. 6.1 Plné rozsahy dynamického vertikalizátora.....	29
Tab. 7.1 Výsledky t-testu pri náklone vpravo.....	36
Tab. 7.2 Výsledky t-testu pri náklonoch.....	36
Tab. 7.3 Výsledky t-testu percentuálneho dosiahnutia plného rozsahu pri náklone vľavo.....	37
Tab. 7.4 Výsledky t-testu percentálneho dosiahnutia plného rozsahu.....	37
Tab. 7.5 Výsledky t-testu priemernej rýchlosti na ose X.....	38
Tab. 7.6 Výsledky t-testu priemernej rýchlosti na osi Y.....	39
Tab. 7.7 Výsledky t-testu najväčšieho dosiahnutého zrýchlenia na ose Y.....	39
Tab. 7.8 Výsledky t-testu najväčšieho dosiahnutého zrýchlenia na osi X.....	40
Tab. 7.9 Výsledky t-testu percentuálneho dosiahnutia linearitu pohybu.....	41

1 ÚVOD

Zdravotné postihnutie je pre človeka ťažkým bremenom, či už s ním žije od narodenia, alebo si ho privodil zranením počas života. Ľudia so zdravotným postihnutím, či už fyzickým alebo psychickým, sú častokrát silní vôľou, no každý z nich vyžaduje zdravotnú starostlivosť a rehabilitáciu, aby sa dokázal zapojiť do bežného života. Preto im rehabilitačný proces, ktorý je časovo veľmi náročný, treba čo najviac spríjemniť rôznymi pomôckami, napríklad interaktívnymi hrami na báze biofeedbacku.

Pacienti so zdravotným postihnutím dolných končatín v dôsledku poranenia, či vrodenej vady mozgu, miechy alebo chrbtice, paraplegici, tetraplegici, pacienti so sklerózou multiplex, svalovou distrofiou, pacienti trpiaci poruchou rovnováhy, ale aj tí, ktorí sú dlhodobo nútení ležať na lôžku v dôsledku úrazu, alebo rôznych druhov operácií, mávajú závažné zdravotné problémy spôsobené ich čiastočnou alebo úplnou nehybnosťou. Medzi tieto problémy patria napríklad preležaniny, ochabnutie svalstva pohybového aparátu, ale aj problémy s trávením, peristaltikou čriev a problémy močových ciest a vylučovania. Spomínané zdravotné problémy sú hlavným podnetom na čo najefektívnejšiu rehabilitáciu pacienta pomocou fyzioterapie. Vo fyzioterapii sa u tejto skupiny pacientov najčastejšie využíva takzvaná vertikalizácia, čo je postavenie pacienta do polohy v stojí.

Vertikalizáciu umožňuje dynamický vertikalizátor, ktorým sa budem zaoberať v mojej bakalárskej práci. Mojim cieľom v prvej časti práce je skonštruovať prototyp bezdrôtového interface medzi vertikalizátorom a obslužným počítačom pomocou vývojovej dosky Arduino Leonardo Pro Micro a komponentov potrebných na snímanie polohy a bezdrôtový prenos údajov do počítača. Vďaka tomuto spojeniu bude možné ovládať klávesnicové šípky a tým hrať terapeutickú hru na počítači. Zároveň sa budú do vývojového prostredia Matlab posielat' polohové dáta, ktoré budú neskôr spracované a vyhodnotené. V druhej časti bakalárskej práce je mojim cieľom vytvoriť program, prostredníctvom ktorého bude možné namerané polohové dáta analyzovať v grafickom prostredí. Hodnotený bude rozsah, linearita a rýchlosť pohybu. Získané výsledky budú následne štatisticky spracované.

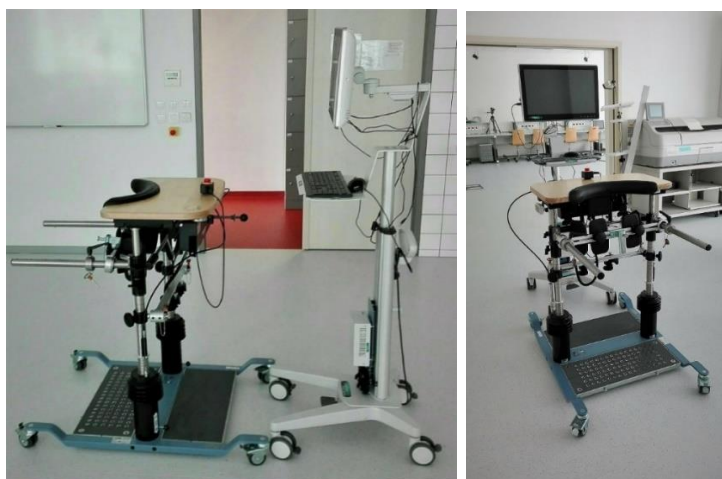
2 TERAPEUTICKÉ HRY V PROCESE REHABILITÁCIE DYNAMICKÝM VERTIKALIZÁTOROM

Cieľom tejto kapitoly je zoznámiť sa s dynamickým vertikalizátorom, prístrojom BALANCE-Trainer a využitím terapeutických hier v procese rehabilitácie.

2.1 Dynamický vertikalizátor

Vertikalizátor je prístroj, ktorý je schopný uviesť pacienta do vertikálnej polohy, teda do stoja. Statický vertikalizátor slúži na postavenie pacienta, dynamický vertikalizátor umožňuje okrem postavenia pacienta aj pacientov pohyb do všetkých strán, aby mohol trénovať a posilňovať svoju stabilitu.

Prístroj **BALANCE-Trainer** je statický a dynamický vertikalizátor (Obr. 2.1.1). Pacienti so zdravotným postihnutím dolných končatín, ktorí sa nemôžu sami postaviť, prípadne majú schopnosť postaviť sa obmedzenú, sú pomocou rôznych typov nastavcov upevnení ku konštrukcii prístroja na úrovni panvy a kolien. Prístroj tak umožňuje bezpečný vzpriamený postoj pacienta. Váha pacienta sa rozloží na prístroj a tak sa pacient môže sústrediť na pokyny fyzioterapeuta a samotné cvičenie bez obáv o svoju stabilitu a strachu z pádu. [1] [2]



Obrázok 2.1.1. Dynamický vertikalizátor BALANCE-Trainer z rôznych uhlov

Prístroj BALANCE-Trainer sa skladá zo súčastí na obrázku 2.1.2.



Obrázok 2.1.2 Popis prístroja BALANCE-Trainer

Základ terapeutického prístroja BALANCE-Trainer tvorí plocha na státie (1), základový rám, podpery kolien (2) a podpery bedier (3). Prístroj je eventuálne možné vybaviť aj elektrickým zdvíhacím systémom, ktorý je tvorený jednotkou s navíjačom pásu, transformátorom a elektrickým káblom.

Prístroj umožňuje nastavenie jednotlivých častí tak, aby vyhovovali potrebám jednotlivých pacientov. Je možné nastaviť polohu a výšku terapeutického stola (4), bedrových a kolenných podpier, madiel (5) a nastaviteľná je aj funkcia Balance (6), ktorá slúži na tréning rovnováhy.

Funkcia umožňuje tri možné stavy, ktorými určuje rozsah pohybu prístroja vo všetkých smeroch, teda vľavo, vpravo, vpred, vzad:

- 0° - funkcia Balance je blokováná,
- 6° - funkcia Balance je povolená s obmedzeným rozsahom pohybu,
- 12° - funkcia Balance je povolená v plnom rozsahu pohybu. [1][2]

2.2 Terapeutické hry

Proces rehabilitácie je pre pacientov často psychicky náročný a preto sa využívajú a hľadajú formy, ako pacientom tento proces čo najviac spríjemniť. Jednou z takýchto foriem je hranie terapeutických hier. Rehabilitácia sa tak stane pre pacienta znesiteľnejšou a zábavnejšou.

Ako príklad môžeme uviesť štúdiu, kedy tím vedeckých pracovníkov a lekárov testoval terapeutické hry v procese rehabilitácie u detí s detskou mozgovou obrnou. Terapeutická hra mala zlepšiť ich balans. Výsledkom štúdie bolo potvrdenie, že terapeutické hry sú naozaj efektívne, zlepšujú zdravotný stav a ich výhodou je, že rehabilitácia sa môže odohrávať v prostredí domova. [3]

Okrem spríjemnenia rehabilitácie pacienta slúžia terapeutické hry aj na kontrolu stavu, respektíve zlepšenia stavu pacienta. Hraním hry pacient opisuje istú trajektóriu, ktorá je vykreslená do grafu. Lekár následne graf posudzuje a vyhodnocuje stav pacienta.

2.3 Ovládanie terapeutických hier

Hranie, respektíve ovládanie terapeutických hier v procese rehabilitácie dynamickým vertikalizátorom využíva pohyb polohového senzoru v troch osiach. Tento senzor je upevnený na konštrukcii prístroja, v ktorom pacient stojí a nakláňa ho do strán pohybom tela, čím simuluje stláčanie klávesnicových šípok a tak umožňuje ovládanie hry.



Obrázok 2.3.1 Umiestnenie prototypu snímača polohy

2.4 Hodnotenie rozsahu pohybu z nameraných dát

Okrem ovládania hier slúži snímač polohy vertikalizátora k diagnostike. Lekár vyhodnocuje spracované namerané dáta zo snímača a vďaka tomu určí rozsah pohybu pacienta, jeho stabilitu. Namerané dáta sú spracované do grafov a tabuliek, ktoré sú ľahko čitateľné a umožňujú prehľad výsledkov rehabilitácie pacienta. V grafoch môže byť vykreslená trajektória pohybu pacienta porovnaná s trajektóriou, keby bol pohyb vykonaný v plnom rozsahu prístroja. V tabuľkách môžeme nájsť napríklad maximálny náklon pohybu do strán.

3 PROTOTYP SNÍMAČA POLOHY

Cieľom bakalárskej práce je skonštruovať a naprogramovať prototyp snímača polohy tak, aby bolo možné prostredníctvom pohybových dát z akcelerometra alebo gyroskopu ovládať šípky klávesnice a vďaka tomu hrať terapeutické hry. Namerané dáta budú následne spracované a vyhodnotené a ich prehľad zobrazený vo vhodnom grafickom prostredí.

Hardwarová časť zariadenia nie je súčasťou počítača, pretože musí byť pripevnená na konštrukcii dynamického vertikalizátora. Prototyp by tak mal byť skonštruovaný z vhodnej vývojovej dosky a komponentov slúžiacich na snímanie pohybu a prenos údajov do počítača. Ako vývojová doska bola použitá doska Arduino a komponenty modul akcelerometra s gyroskopom a Bluetooth modul.

Pred samotným popisom súčiastok je potrebné oboznámiť sa s princípom fungovania akcelerometra a gyroskopu, keďže tieto zariadenia sú najdôležitejšou časťou umožňujúcou získanie polohových dát.

3.1 Akcelerometer

Akcelerometer je elektromechanické zariadenie, ktoré meria zrýchlenie a hodnotu zrýchlenia prevádza na hodnotu elektrického napätia.

Zrýchlenie je definované ako prvá derivácia rýchlosti podľa času, teda zjednodušene môžeme povedať, že zrýchlenie predstavuje zmenu rýchlosti za jednotku času. Delí sa na zrýchlenie statické a dynamické. **Dynamické zrýchlenie** vzniká zmenou rýchlosti pohybu objektu, na ktorý je senzor pripojený a udáva smer, ktorým sa objekt pohybuje v priestore. **Statické zrýchlenie** vzniká v dôsledku tiažového zrýchlenia Zeme a umožňuje výpočet uhla vychýlenia vzhľadom k povrchu Zeme. Základnou jednotkou zrýchlenia je m/s^2 . [7]

Princíp akcelerometra je založený na meraní výchyliek pružne hmotného telesa, ktoré sa nachádza vnútri akcelerometra. Zrýchlenie je priamo úmerné vonkajšej sile, čo v našom prípade odráža intenzitu vykonaného pohybu a preto je vhodné akcelerometer použiť na meranie pohybovej aktivity a rozsahu pohybu.

Každý akcelerometer má určitú **citlivosť**, ktorá sa dá definovať ako výstupné napätie snímača pri meraní určitej sily, ktorá je vyjadrená v g ($1g = 9,81 \text{ m/s}^2$). Amplitúda

a úroveň výstupného striedavého napätia zodpovedá amplitúde vstupného (meraného) signálu. Čím má akcelerometer vyššiu citlivosť, tým väčšiu zmenu pozorujeme v signáli meranom pri danej zmene rýchlosti. V našom prípade nemusí mať akcelerometer vysokú citlivosť, keďže meriame výraznú zmenu pohybu s väčšou amplitúdou a nie zmenu na úrovni vibrácií, ktorých amplitúda je malá.

Ďalšou dôležitou charakteristikou akcelerometra je jeho **frekvenčný rozsah**, ktorý udáva počet dôb za sekundu, v ktorých je akcelerometer schopný merať zrýchlenie spoľahlivo. Mal by zodpovedať predpokladanému rozsahu meraných frekvencií.

V obvodoch se môže indukovať širokopásmové alebo spektrálne frekvenčné **rušenie**, ktoré môže narúšať priebeh signálu. Širokopásmové rušenie je také rušenie, pri ktorom sa prekrýva frekvenčné spektrum užitočného signálu s frekvenčným spektrom šumu a frekvenčné rušenie postihuje určité frekvencie. Pri nízkych frekvenciách je potrebné signál z tohto dôvodu filtrovať, pretože je zaťažený rušením. Vyššie frekvencie tento **šum** výrazne neovplyvňuje, pretože platí, že s rastúcou frekvenciou šum klesá.

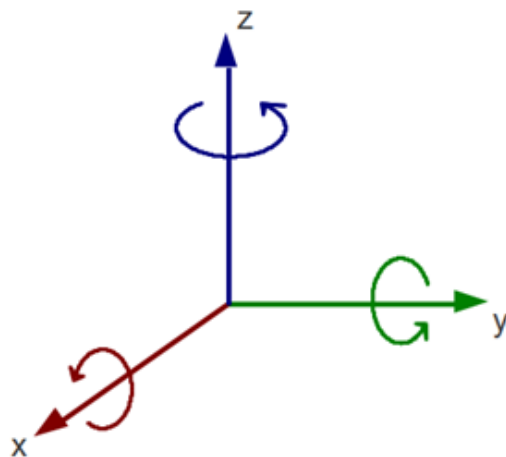
Dynamický rozsah je interval hodnôt amplitúdy snímanej veličiny, kedy sa prístroj nepoškodí. Udáva sa v násobkoch g a jeho šírka by mala byť väčšia než je predpokladaný rozsah amplitúdy meraného signálu.

Hmotnosť akcelerometra by mala byť veľa násobne nižšia ako hmotnosť meraného objektu. Dôvodom je to, že pokiaľ by boli hmotnosti blízke, mohlo by dôjsť k ovplyvneniu vlastného kmitočtu a zrýchlenia meraného objektu. S rastúcou hodnotou podielu hmotností sa miera ovplyvnenia kmitočtu a zrýchlenia znižuje. Tento podiel by mal byť väčší ako 10. [7] [8]

3.2 Gyroskop

Gyroskop sa používa najmä pri navigácii. Umožňuje meranie uhlovej rýchlosti a orientácie v priestore. Gyroskopy sa delia na rotačné, vibračné a optické.

Rozsah merania gyroskopu určuje maximálnu uhlovú rýchlosť merateľnú daným senzorom. Táto rýchlosť je zvyčajne vyjadrená v stupňoch za sekundu ($^{\circ}/s$). V prípade gyroskopu s tromi osami je priestorová orientácia telesa špecifikovaná parametrami rotácie okolo osí (x, y, z), čo môžeme vidieť na obrázku 3.1.1.



Obrázok 3.2.1 Znázornenie osí a ich rotácií modulu MPU-6050

Šírka pásma udáva koľko meraní môže byť uskutočnených v priebehu jednej sekundy, z čoho vyplýva, že jej jednotkou bude Hz.

Mieru šumu gyroskopu udáva **náhodná uhlová prechádzka** (AWR – Angular Random Walk) a jej jednotkou je stupeň za sekundu ($^{\circ}/s$). [10]

4 HARDWARE PROTOTYPU SNÍMAČA POLOHY

Cieľom tejto kapitoly je priblížiť vývojovú dosku Arduino a prototyp snímača polohy zloženého z dosky Arduino Leonardo Pro Micro, akcelerometra a Bluetooth modulu.

4.1 Arduino

Arduino bolo vyvinuté v roku 2005 skupinou ľudí z talianskeho Interaction Design Institute a bolo určené najmä pre študentov, ktorí na svoju prácu potrebovali jednoduchý a lacný vývojový set. Tvorcovia sa ho neskôr rozhodli poskytnúť celosvetovo a to tak, že umožnili zdieľať všetky schémy, software a návody bezplatne, jedná sa teda o Open Source projekt. Používatelia si musia zaobstarat iba vývojovú dosku. Vývojových dosiek je niekoľko typov a je na užívateľovi, ktorý typ si zvolí podľa projektu, na ktorom sa chystá pracovať. [4]

Základom väčšiny dosiek Arduino je procesor firmy Atmel, ktorý dotvárajú ďalšie elektronické komponenty. Jednotlivé typy dosiek sa líšia práve použitými komponentami, ich rozložením a dizajnom. Väčšina dosiek obsahuje okrem hlavného čipu ešte čip, ktorý zaobstaráva komunikáciu s počítačom. [4] [5]

4.1.1 Vývojové dosky Arduino

Dôvodom použitia vývojovej dosky Arduino je jej cenová dostupnosť a najmä praktickosť. Skonstruovanie polohového senzora je vďaka nej jednoduchšie. Dôležitým parametrom, ktorý musí vybraná vývojová doska Arduino spĺňať, je možnosť ovládania klávesnice, vďaka čomu bude možné ovládanie počítačovej hry.

Medzi jednotlivé typy Arduina patrí **Arduino Mini**, ktoré je najmenším Arduinom a práve to zabezpečuje úsporu miesta. Ďalším typom je **Arduino Nano** a **Arduino Micro**, ktoré je o niečo väčšie. Výhodou Arduina Micro je, že môže zastávať úlohu klávesnice alebo myši pre počítač bez preprogramovania prevodníku. **LilyPad Arduino** je prispôbené na uchytenie na textil, vďaka tomu, že spojenie Arduina s perifériou je tvorené vodivými niťami. **Arduino Uno** je v tejto dobe najpoužívanejšou doskou s procesorom ATmega328. Z Arduina Uno boli vyvinuté dosky **Arduino Ethernet**, ktoré namiesto klasického USB portu využíva Ethernet, vďaka čomu umožňuje

pripojenie priamo na sieť a **Arduino Bluetooth**, ktoré využíva bezdrôtovú komunikáciu prostredníctvom Bluetooth modulu. **Arduino Leonardo** sa od Arduina Uno líši použitím čipu ATmega32u4, ktorý je použitý aj u Arduina Micro. **Arduino Mega2560** je v podstate predĺžené Arduino Uno, má viac pinov a väčšie a výkonnejšie čipy. Prostredníctvom **Arduina Esplora** je možné vytvoriť vlastnú konzolu na ovládanie hier. Už z názvu **Arduino Robot** vyplýva, že toto Arduino je najvhodnejším pri vytvorení vlastného robota. **Arduino Intel Galileo** je doska, ktorá vznikla v spolupráci s firmou Intel. Ďalšie dostupné dosky sú **Arduino Fio**, **Arduino Yún**, **Arduino Due** a **Tre**. [4]

4.2 Arduino Leonardo Pro Micro

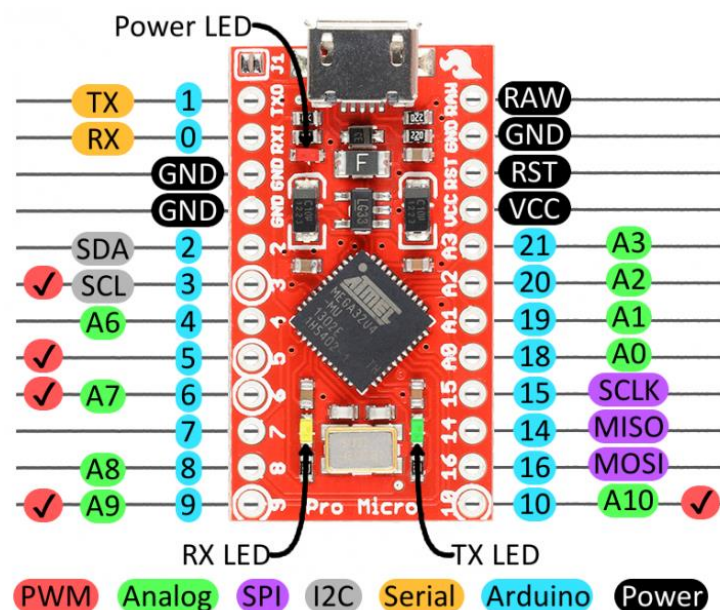
Vývojová doska, ktorá je použitá pri tejto bakalárskej práci, je doska **Arduino Leonardo Pro Micro**, ktorú môžeme vidieť na obrázku 4.2.1.



Obrázok 4.2.1 Arduino Leonardo Pro Micro

Arduino Leonardo Pro Micro je miniatúrna vývojová doska, na ktorej je umiestnený 8 bitový **mikrokontrolér AVR ATmega32u4**, ktorý je taktovaný na kmitočte 16 MHz. Vďaka integrovanému P/S prevodníku je umožnená komunikácia prostredníctvom USB bez použitia dodatočných komunikačných prevodníkov.

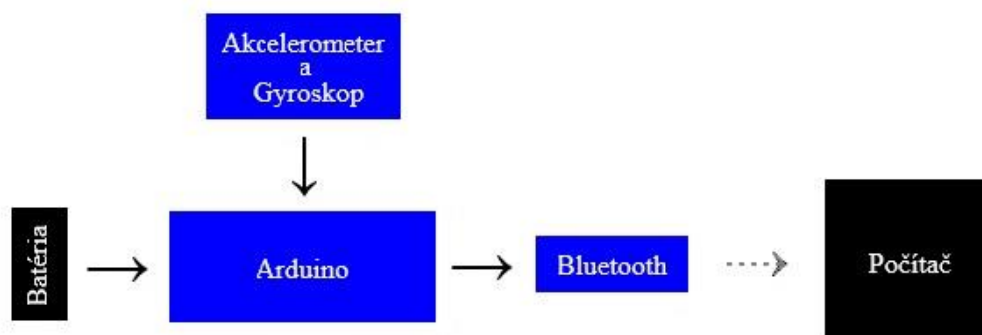
Leonardo Pro Micro obsahuje 12 I/O pinov, z čoho 9 pinov je analógových, jedno sériové rozhranie, I2C zbernica a SPI zbernica. Ich rozloženie na doske môžeme vidieť na obrázku 4.2.1.



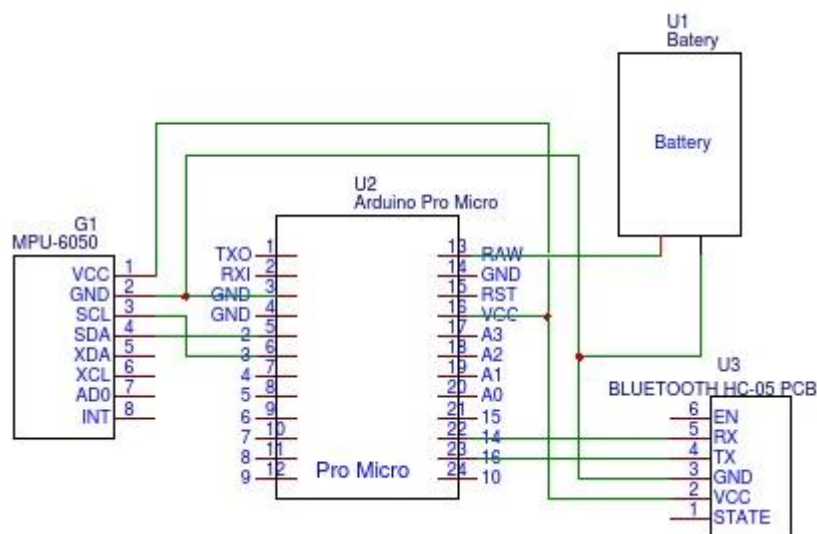
Obrázok 4.2.2 Rozloženie pinov na doske Arduino Leonardo Pro Micro

Na programovanie a prípadné napájanie slúži **Micro USB konektor**. Externé napájanie 6 V ~ 9 V je možné priviesť na vývod RAW. Na napájanie prototypu snímača bola použitá 9 V batéria. Dôvodom jej použitia bola široká dostupnosť a pre používanie prototypu snímača polohy bola plne dostačujúca.

V mojej práci boli použité piny GND, RAW a VCC, ktoré slúžili na uzemnenie a napájanie napätím 9 V z batérie, ktoré bolo následne usmernené a regulované. Ďalšie použité boli piny 2 a 3, ktoré slúžili na komunikáciu s akcelerometrom a piny 14 a 16, ktoré slúžili na komunikáciu s Bluetooth modulom a definované budú v nasledujúcich kapitolách. Zapojenie komponentov zobrazuje obrázok 4.2.3 a 4.2.4.



Obrázok 4.2.3 Schéma zapojenia komponentov



Obrázok 4.2.4 Schéma zapojenia prototypu snímača polohy

4.3 Snímač polohy

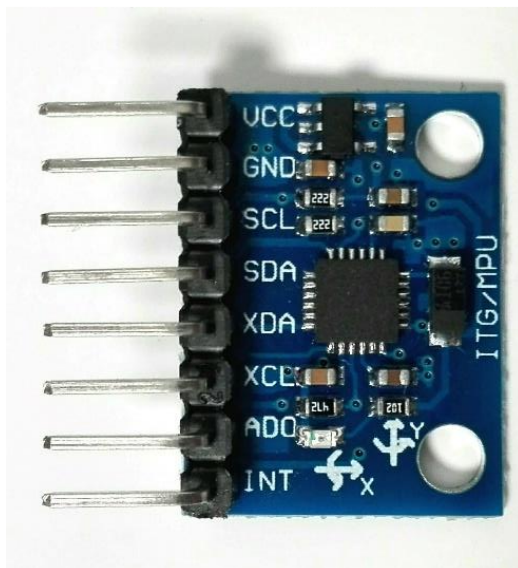
Snímačom polohy je **modul MPU-6050**, ktorý sa skladá z dvoch zariadení, a to akcelerometra a gyroskopu.

Akcelerometer MPU-6050 je trojosí akcelerometer (x, y, z). Zrýchlenie v smere jednej zo snímaných osí spôsobí posun na zodpovedajúcej kontrolnej ploche a následne ho detekujú kapacitné snímače. Zariadenie položené na rovnom povrchu bude mať nulové hodnoty na osi x a y , na osi z bude hodnota rovná 1. Plný rozsah digitalizovaného výstupu môže byť upravený na $\pm 2g$, $\pm 4g$, $\pm 8g$ alebo $\pm 16g$. Použitý bol rozsah $\pm 2g$, čo znamená, že maximálne zrýchlenie modulu bolo $2g$. Na nehybne stojaceho človeka pôsobí sila $1g$. Pri pohyboch vykonávaných na dynamickom vertikalizátore sa táto sila mierne zvýši vďaka dynamike pohybu pacienta, ale kvôli odporu pružín vertikalizátora neprekročí nastavený rozsah. Použitý rozsah bol teda z hľadiska použitia modulu na zaznamenanie dostatočujúci, čo následne dokázala absencia orezaných špičiek v zaznamenanom signále, ktoré by prezradzovali saturáciu senzoru v mieste orezania. [9]

GY-521 obsahuje trojosí gyroskop, ktorý sníma uhlovú rýchlosť. Plný rozsah stupnice je možné naprogramovať na hodnoty $\pm 250^\circ/s$, $\pm 500^\circ/s$, $\pm 1000^\circ/s$ a $\pm 2000^\circ/s$. Použitý bol rozsah $\pm 250^\circ/s$, čo odpovedá maximálnej uhlovej rýchlosti $250^\circ/s$ a teda vyhovoval potrebnému rozsahu. GY-521 je zložený z troch vibračných MEMS gyroskopov, vďaka ktorým je možná rotácia okolo osí (x, y, z). Pri natáčaní senzoru pozdĺž niektorej z osí spôsobuje Coriolisova sila vibrácie, ktoré sú následne snímané.

Následne je signál zosilnený, demodulovaný a filtrovaný na napätie, ktoré je úmerné zrýchleniu získaného z akcelerometra. Vzorkovacia frekvencia signálu je v rozmedzí od 3,9 Hz do 8 000 Hz a zariadenie pracuje pri prúde 3,6 mA. [6] [9]

Realizáciu snímača je vidieť na Obrázku 4.3.1.



Obrázok 4.3.1 Akcelerometer + Gyroskop MPU-6050

4.3.1 Digital motion procesor a zbernica I2C

Modul MPU-6050 obsahuje aj **integrovaný obvod DMP** (digital motion procesor) s funkciou zastupujúcou obe zariadenia – akcelerometer a gyroskop. Vďaka DMP je možné nechať z akcelerometrických a gyroskopických dát vypočítať náklon a sklon priamo na doske modulu a nezaťažovať výpočtom procesor Arduino.

I2C je zbernica, ktorá umožňuje mať na dvoch vodičoch pripojených až 128 zariadení, pričom jeden vodič udáva takt komunikácie - SCL (clock line) a druhý prenáša dáta – SDA (data line). Pri tomto type komunikácie je vždy jedno zo zariadení riadiace – master, v našom prípade Arduino Leonardo Pro Micro a ostatné sú riadené – slave, v našom prípade modul MPU-6050. [5]

Na pripojenie s Arduino Leonardo Pro Micro boli použité piny SCL a SDA, na napájanie pin VCC a na uzemnenie pin GND.

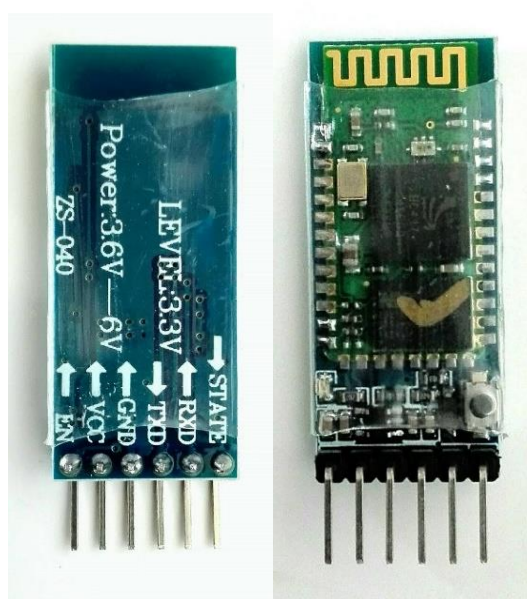
Piny, na ktoré bola pripojená zbernica I2C na Arduino Leonardo Pro Micro, sú označené číslom 2 a 3, pričom 2 zodpovedá pinu SDA a 3 pinu SCL. Pin 3 (SCL)

z Arduina bol napojený na pin SDA z akcelerometra a pin 2 (SDA) z Arduina bol napojený na pin SCL z akcelerometra.

4.4 Bezdrôtový prenos údajov do počítača

V práci bol použitý bezdrôtový prenos údajov do počítača z dôvodu modernizácie terapeutického prístroja BALANCE-Trainer. Vďaka možnosti prenášania údajov bezdrôtovo nie je nutné umiestniť prístroj na presne stanovené miesto v miestnosti, ktoré by mohlo byť limitované dĺžkou prenosového kábla a taktiež sa obmedzí možnosť nechceného odpojenia prístroja od počítača v dôsledku odpojenia prenosového kábla.

Na prenos údajov bol použitý Bluetooth modul HC-05, ktorý má dostatočný rozsah pokrytia a zabezpečuje rýchlu komunikáciu medzi zariadeniami s podporou Bluetooth.



Obrázok 4.4.1 Bluetooth modul HC-05

Bluetooth modul HC-05 slúži na bezdrôtový prenos údajov medzi Arduino a ďalším zariadením, ktoré podporuje Bluetooth. Modul obsahuje Bluetooth 2.0 a s Arduino komunikuje prostredníctvom sériovej linky s počiatočnou rýchlosťou 9600 baudov. Dosah je obmedzený na vzdialenosť 10 metrov vo voľnom priestranstve. Napájacie napätie by malo byť v rozsahu od 3,3 do 6 V. Pri napätí 5 V sa prúdový odber pohybuje okolo hodnoty 2 mA v kľude a pri prenose údajov dosahuje najviac 40 mA. [11]

Modul HC-05 sa s doskou Arduino Leonardo Pro Micro spája pomocou 4 pinov, a to RXD, TXD, GND a VCC. Pin RXD z Bluetooth modulu bol pripojený na pin 14 (MISO) z vývojovej dosky a pin TXD z Bluetooth modulu bol pripojený na pin 16 (MOSI) z vývojovej dosky Arduino.

Piny 14 a 16 na vývojovej doske Arduino Leonardo Pro Micro obsahujú SPI (Serial Peripheral Interface), ktoré sa používa na komunikáciu medzi riadiacimi mikroprocesormi a ostatnými integrovanými obvodmi. Na pine 14 sa nachádza MISO (Master In, Slave Out), teda master je vstup a slave výstup a slúži ako TXD. Na pine 16 sa nachádza MOSI (Master Out, Slave In), teda master výstup a slave vstup a slúži ako RXD. Pin VCC slúži na napájanie a pin GND na uzemnenie Bluetooth modulu. Rozloženie pinov môžeme vidieť na obrázku 4.4.1. [11]

4.4.1 Piny

Modul HC-05 má 6 pinov, čo môžeme vidieť na obrázku 4.4.1. Ich prehľad a popis nájdeme v Tabuľke 4.1.

Tabuľka 4.1 Piny Bluetooth modulu HC-05

Pin	Popis	Funkcia
→ STATE	Pripojenie na LED	Signalizuje aktivitu senzora
← RXD	Sériový prijímač signálu	Prijíma signál, pripája sa na TXD pin
→ TXD	Sériový vysielač signálu	Vysiela signál, pripája sa na RXD pin
← GND	Ground	Uzemňuje zariadenie
← VCC	+ 5 V	Privádza napätie
← EN	+ 3,3 V	Zadávanie príkazov

5 SOFTWARE PROTOTYPU SNÍMAČA POLOHY

Cieľom tejto kapitoly je priblížiť programovanie samotného prototypu snímača polohy, ovládanie hier a meranie dát.

5.1 Programovanie prototypu snímača polohy

Vývojové prostredie Arduina sa volá **Arduino IDE** a tvorí ho textový editor na písanie kódu, priestor na správy, textová konzola, lišta s príkazmi a rôzne iné ponuky. Jeho úlohou je pripojiť sa na hardware Arduina, nahrať doň program a komunikovať s ním.

Samotný skript programu sa skladá z troch častí. Prvá časť slúži na definovanie premenných a zahrnutie knižníc, ktoré slúžia na písanie základných operácií. V druhej a tretej časti musí užívateľ definovať dve funkcie, a to *setup* a *loop*. Funkcia *setup* sa spúšťa len raz a to na začiatku programu, pričom slúži na nastavenie parametrov. Funkcia *loop* sa spúšťa periodicky po dobu, kedy je mikroprocesor pripojený k napájaniu. [5]

Pri programovaní **modulu MPU-6050** v Arduino IDE sa používa knižnica *Wire.h*, ktorá bola hneď v úvode zahrnutá do skriptu. Ďalej sú v úvode definované premenné, do ktorých sa ukladajú dáta z akcelerometra a gyroskopu.

V časti **void setup** boli použité funkcie *Wire.begin*, *Wire.beginTransmission*, *Wire.write* a *Wire.endTransmission*. Funkcia *Wire.beginTransmission (MPU_addr)* začína komunikáciu s MPU-6050 a funkcia *Wire.endTransmission (true)* naopak túto komunikáciu ukončí. Funkcia *Wire.begin()* pripája zariadenie ku zbernici I2C a funkcie *Wire.write (0x6B)* a *Wire.write (0)* slúžia na odosielanie informácií práve po tejto zbernici, nastavenie počtu bitov a ich dĺžku a nastavenie prvej hodnoty na 0.

Časť **void loop** opäť zahŕňa funkcie *Wire.beginTransmission (MPU_addr)*, *Wire.write (0x3B)*, *Wire.endTransmission (false)* a ďalej funkciu *Wire.requestFrom (MPU_addr,14,true)*, ktorá vyžaduje dáta od slave a určuje očakávaný počet bytov a *Wire.read()*, ktorá číta hodnoty z I2C a zapisuje do definovaných premenných.

Vzorkovacia frekvencia zaznamenávania dát bola stanovená na 25 Hz, čo znamená, že každých 40 ms sa dáta zaznamenajú.

Programovanie **Bluetooth modulu HC-05** v úvode zahŕňa knižnicu *SoftwareSerial.h*, z ktorej bol vybraný *SoftwareSerial bluetooth (TX, RX)*, čím bol definovaný prijímač RXD a vysielač TXD. RXD bol priradený k pinu 14 a TXD k pinu 16. V časti **void setup** bola nastavená počiatočná rýchlosť na 9600 baudov prostredníctvom funkcie *bleutooth.begin (9600)*. V časti **void loop** boli pomocou funkcie *bluetooth.print (AcX, DEC)* uložené a zapísané dáta do vopred definovaných premenných (*AcX, AcY, AcZ, GyX, GyY, GyZ*).

5.2 Ovládanie klávesových šípok

Ovládanie šípok bolo uskutočnené dvomi rôznymi spôsobmi, a to simuláciou v programe Arduino IDE a simuláciou v programe Matlab R2017a.

5.2.1 Simulácia v programe Arduino IDE

Arduino Leonardo Pro Micro umožňuje ovládanie myši a počítačovej klávesnice. Na ovládanie klávesnice je potrebné v **Arduino IDE** zahrnúť knižnicu *Keyboard.h*, ktorá sprostredkúva komunikáciu medzi Arduinom a počítačovou klávesnicou. V časti **void setup** bola použitá funkcia *Keyboard.begin()*, ktorá inicializuje pripojenie emulátora klávesnice k počítaču. V časti **void loop**, bola použitá funkcia *Keyboard.press(d)*, ktorá simuluje stlačenie klávesy „d“ na počítačovej klávesnici a funkcia *Keyboard.release(d)*, ktorá simuluje pustenie klávesy „d“. Rovnaké funkcie boli použité aj na stlačenie a pustenie klávesy „a“, prípadne šípok „dolu“, „hore“, „vpravo“, „vľavo“.

Keďže použitá hra vyžadovala stláčanie kláves „a“ a „d“, simulovanie bolo použité práve na tieto klávesy. Klávesa „a“ značí pohyb vľavo a klávesa „d“ vpravo. Na udanie **smeru** pohybu, respektíve stlačenia klávesy bol použitý **senzor pohybu** – akcelerometer + gyroskop MPU-6050, konkrétne Y – os gyroskopu. Na prepočet hodnôt bola použitá rovnica:

$$vy = \frac{GyY+15}{2}, \quad (1)$$

kde vy je prepočítaná hodnota osy GyY a GyY je Y – os gyroskopu. K hodnote GyY bola pripočítaná hodnota 15 a výsledná hodnota bola delená hodnotou 2 z dôvodu získania menších výchyliek od nulovej línie. Na simulovanie stlačenia kláves pomocou hodnôt vy bol použitý cyklus *if*, pričom ak bola hodnota vy väčšia ako 0, stlačená bola klávesa „d“ a zároveň pustená klávesa „a“ a ak bola hodnota vy menšia ako 0, stlačená bola klávesa „a“ a pustená klávesa „d“, čím sa zaručil plynulý pohyb šípky v hre.

Obdobným spôsobom bolo naprogramované aj stláčanie klávesových šípok, kedy na pohyb vľavo a vpravo bola použitá už uvedená rovnica (1) a cyklus *if* a na pohyb hore a dolu bola použitá nasledovná rovnica:

$$vx = \frac{GyX}{15}, \quad (2)$$

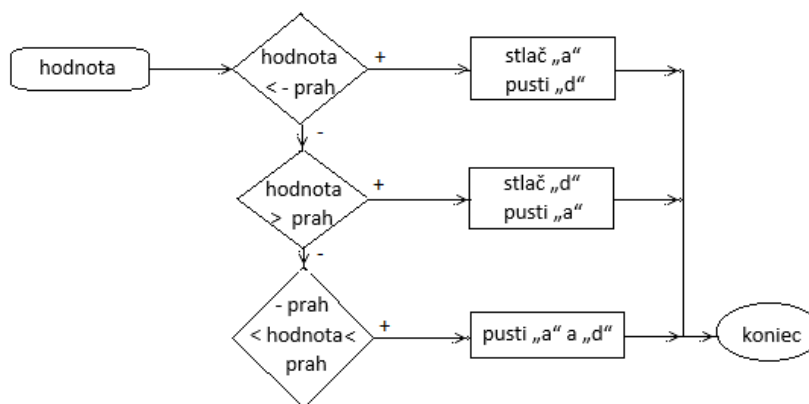
kde vx je prepočítaná hodnota osy GyX a GyX je X – os gyroskopu. Hodnota GyX bola delená hodnotou 15 z dôvodu získania menších výchyliek od nulovej línie. Na simulovanie stlačenia šípok bol taktiež použitý cyklus *if*, pričom ak bola hodnota vx väčšia ako 0, stlačená bola šípka hore a zároveň pustená šípka dolu a ak bola hodnota vx menšia ako 0, stlačená bola šípka dolu a pustená šípka hore.

5.2.2 Simulácia v programe Matlab R2017a

Bezdrôtové ovládanie klávesnice pomocou Arduino IDE sa však nepodarilo, ovládanie bolo funkčné len pri použití USB kábla, v dôsledku čoho bol naprogramovaný ďalší skript, tentokrát v programe Matlab R2017a.

Prijímanie, ukladanie a zobrazovanie dát prebieha vďaka doplnku **ArduinoHardware Support**, ktorý umožňuje komunikáciu medzi vývojovou doskou Arduino, jej komponentami a Matlabom. Do Matlabu boli prijímané dáta z akcelerometra a gyroskopu prostredníctvom funkcie *Bluetooth*, ktorá umožňuje prijímanie dát z Bluetooth modulu. Dáta boli následne prostredníctvom cyklu *while* vykresľované a ukladané v reálnom čase, pričom sa prepočítavali hodnoty GyY , teda hodnoty z Y – osy gyroskopu na upravené hodnoty vy , ktoré slúžili na simuláciu stláčania kláves. Prepočet na hodnoty vy sa uskutočňoval podľa už uvedeného vzorca (1).

Simulácia stláčania klávesnice sa v Matlabe uskutočňuje pomocou **java.awt.Robot**, čo je emulátor ľudskeho stlačenia klávesy, prípadne myši virtuálnym robotom. Na vyhodnotenie momentu, kedy mala byť stlačená klávesa „a“ pri smere vľavo a klávesa „d“ pri smere vpravo bol použitý cyklus *if – elseif – else – end*. V časti *if* bola použitá podmienka ($hodnota < -prah$) (pričom premenná *hodnota* predstavovala poslednú hodnotu vektoru čísel **vy**, *prah* bol nastavený na hodnotu 1000) a predstavovala stlačenie klávesy „d“ a zároveň pustenie klávesy „a“ pomocou príkazov *robot.keyPress (java.awt.event.KeyEvent.VK_D)* a *robot.keyRelease (java.awt.event.KeyEvent.VK_A)*. V časti *elseif* bola použitá podmienka ($hodnota > prah$) a príkazy *robot.keyPress (java.awt.event.KeyEvent.VK_A)* a *robot.keyRelease (java.awt.event.KeyEvent.VK_D)*, čo simulovalo opačnú situáciu ako v predchádzajúcej časti *if*. V časti *else* boli použité príkazy *robot.keyRelease (java.awt.event.KeyEvent.VK_D)* a *robot.keyRelease (java.awt.event.KeyEvent.VK_A)*, vďaka čomu sa vytvorilo pásmo hodnôt od -1000 do 1000, kedy neboli stláčané žiadne klávesy.

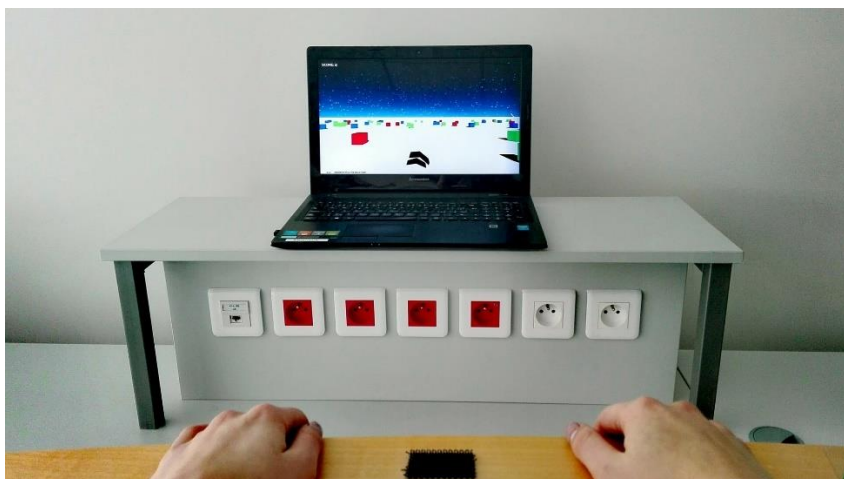


Obrázok 5.2.1 Vývojový diagram simulácie stláčania šípok

5.3 Hra Cube Dodger

Hra určená na ovládanie dynamickým vertikalizátorom by mala byť jednoduchá na ovládanie, ovládanie by malo byť len v smere vpravo, vľavo, vpred a vzad. Na hranie sú teda vhodné hry, ktoré sa ovládajú len stláčaním klávesových šípok, prípadne kláves A, S, D a W.

Hra **Cube Dodger** patrí medzi akčné hry. Hranie je možné na platformách Windows a Android a ovládanie je možné pomocou klávesnice. Cieľom hry je prejsť pomedzi kocky bez toho, aby sa hráč šípkou kociek dotkol (Obr. 5.3.1). Pri použití klávesnice je potrebné stláčať klávesy „A“ a „D“, pričom stlačenie klávesy „A“ značí pohyb šípky doľava a stlačenie klávesy „D“ pohyb doprava. Nastavenia hry umožňujú zvoliť mieru obtiažnosti a rýchlosť pohybu. [12]



Obrázok 5.3.1 Hranie hry Cube Dodger pomocou dynamického vertikalizátora

Pri hraní hry na dynamickom vertikalizátore bola obtiažnosť nastavená na najľahšiu (EASY) a rýchlosť najmenšiu (0,25 x).

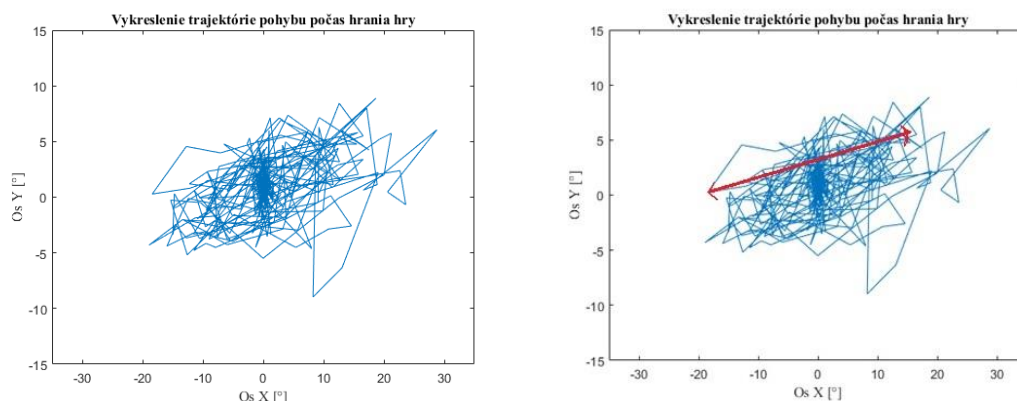
5.4 Meranie polohových dát počas hrania hry

Meranie dát na skupine dobrovoľníkov sa uskutočnilo podľa protokolu merania polohových dát počas hrania hry, ktorý je priložený v prílohách (Protokol merania 1). Meranie prebehlo bez odchýlok od protokolu merania a v Tabuľke 4.2. nájdeme pocity z merania jednotlivých dobrovoľníkov. Skupina dobrovoľníkov bola zložená zo šiestich žien a dvoch mužov, pričom ich priemerný vek bol 21 rokov. Probandi hodnotili **mieru zábavy** na stupnici od 1 do 5, pričom 1 znamená, že meranie nebolo vôbec zábavné a 5, že bolo veľmi zábavné. Hodnotená bola aj **náročnosť**, pričom 1 znamená, že vykonávanie pohybov a hranie hry bolo veľmi jednoduché a 5 znamená, že vykonávanie pohybov a hranie hry bolo veľmi náročné.

Z Tabuľky 5.1 vyplýva, že hranie hry bolo **relatívne náročné**, pretože priemerná hodnota náročnosti bola 3,9. Priemerná hodnota miery zábavy bola taktiež 3,9, takže môžeme skonštatovať, že meranie bolo **dostatočne zábavné**.

Tabuľka 5.1 Hodnotenie merania

Meno a priezvisko	Miera zábavy	Náročnosť
Proband 1 - žena	3	4
Proband 2 - žena	3	4
Proband 3 - žena	3	5
Proband 4 - muž	4	4
Proband 5 - žena	5	3
Proband 6 - muž	5	3
Proband 7 - žena	3	4
Proband 8 - žena	4	4



Obrázok 5.4.1 Trajektória pohybu (vľavo) a vyznačený detail pohybu z jednej strany na druhú (vpravo)

Dáta zaznamenané pomocou polohového senzora počas hrania terapeutickkej hry boli uložené a vykresľované v reálnom čase v programe Matlab R2017a. Na Obrázku 5.4.1 môžeme vidieť vykreslenú trajektóriu pohybu, ktorú opisoval proband počas hrania hry.

5.5 Meranie polohových dát určených na analýzu pohybu

Meranie dát na skupine dobrovoľníkov sa uskutočnilo podľa protokolu merania polohových dát, ktorý je priložený v prílohách (Protokol merania 2). Meranie prebehlo bez odchýlok od protokolu merania. Meraná skupina bola zložená z 10 probandov, pričom muži boli traja a žien bolo 7. Ich priemerný vek bol 21 rokov.

Cieľom tohoto merania bolo získať dáta, ktoré boli určené na **analýzu pohybu**. Analýza spočívala v hodnotení rozsahu pohybu, rýchlosti a linearity pohybu. Tieto dáta boli ďalej spracované postupom, ktorý je ozrejmeneý v nasledujúcej kapitole.

6 ANALÝZA POLOHOVÝCH DÁT

Analýza polohových dát slúži na stanovenie rozsahu, rýchlosti a linearity pohybu človeka na dynamickom vertikalizátore. Spracovanie spočíva v príprave dát, vytvorení algoritmu na spracovanie dát a vytvorení grafického prostredia, pomocou ktorého užívateľ dáta analyzuje. Analýza je vytvorená v programe Matlab R2017a.

6.1 Príprava dát

Pred samotným spracovávaním je potrebné namerané dáta upraviť. Akcelerometrické a gyroskopické „raw data“ je potrebné vydeliť konštantou, vďaka ktorej dostaneme konkrétne hodnoty zrýchlenia a náklonu. Táto konštanta, nazývaná **faktor škálovania**, predstavuje citlivosť senzora v nastavenom rozsahu. U akcelerometra je to 16384 pri nastavenom rozsahu +/- 2g. U gyroskopu je to 131 pri nastavenom rozsahu +/- 250°/s. Na úpravu akcelerometrických dát bol použitý vzorec (4) a na úpravu gyroskopických dát vzorec (5).

$$A_{\text{prepočítané}} = \frac{A_{\text{raw}}}{16384}, \quad (4)$$

$$G_{\text{prepočítané}} = \frac{G_{\text{raw}}}{131}, \quad (5)$$

Na hodnotenie rozsahu, zrýchlenia, rýchlosti a linearity pohybu je potrebné každý zo šiestich nameraných a následne prepočítaných vektorov signálov (**AxP**, **AyP**, **AzP**, **GxP**, **GyP**, **GzP**) rozdeliť na viacero menších vektorov (úsekov) podľa pohybovej aktivity probanda. Preto boli vytvorené vektory na každú spracovávanú časť zvlášť. Na hodnotenie **rozsahu** pohybu boli použité vektory **GxR**, **GyR**, **GzR**, ktoré predstavovali časť signálu zobrazujúcu náklon probanda do strán. Vektory, z ktorých bolo získané **zrýchlenie** a následne **rýchlosť** boli pomenované **AxV** a **AyV** a predstavovali časť, kedy proband pohybom tela opisoval štvorec. Na hodnotenie **linearity** bol použitý vektor **GyL** a predstavoval časť signálu, kedy sa proband snažil čo najrovnejšie prejsť z pravej na ľavú stranu, prípadne naopak.

Os X senzora MPU-6050 predstavuje Y – os pri spracovávaní a os Y senzora predstavuje X – os pri spracovávaní. Táto zmena osí bola spôsobená natočením senzora v prototypu snímača polohy a nepredstavovala problém pri spracovaní dát.

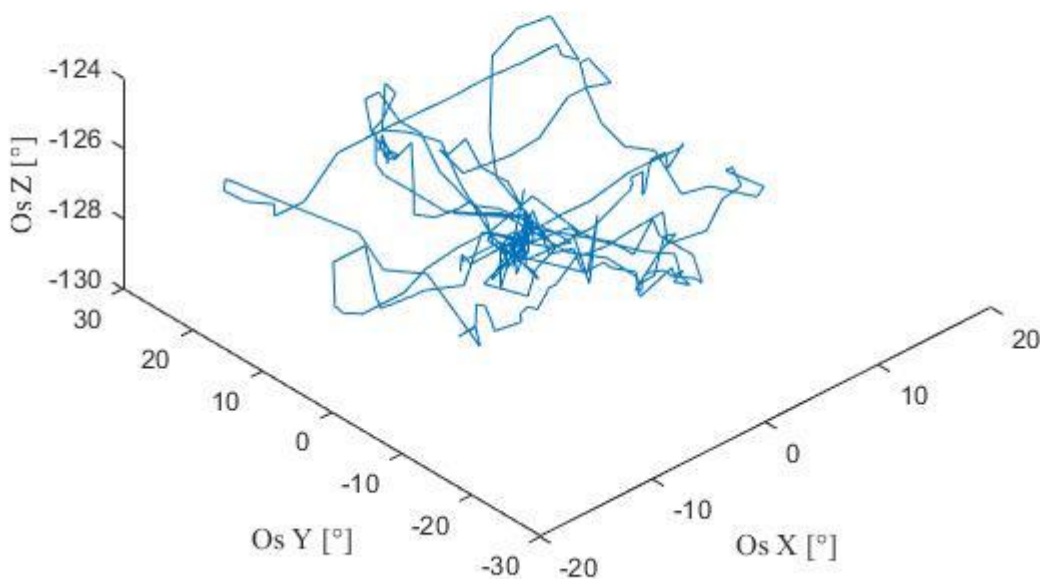
6.2 Spracovanie dát

Spracovanie dát spočíva v stanovení rozsahu, zrýchlenia, rýchlosti a linearitu pohybu probanda vykonávaného na dynamickom vertikalizátore. Na spracovanie sa využívajú už spomínané prepočítané dáta z akcelerometra a gyroskopu. Skript *spracovanie.m* je priložený k práci.

6.2.1 Rozsah pohybu

Rozsah pohybu sa stanovuje z náklonu probanda vpred, vzad, vpavo a vľavo, a teda použité boli namerané a prepočítané gyroskopické dáta. Dáta boli prefiltrované pomocou mediánového filtra s dĺžkou okna 3 vzorky, čím sa zabezpečilo, že najväčšie dosiahnuté hodnoty nebudú získané z nechceného pohybu. Ako prvá bola vykreslená trajektória pohybu v 3D (Obrázok 6.2.1).

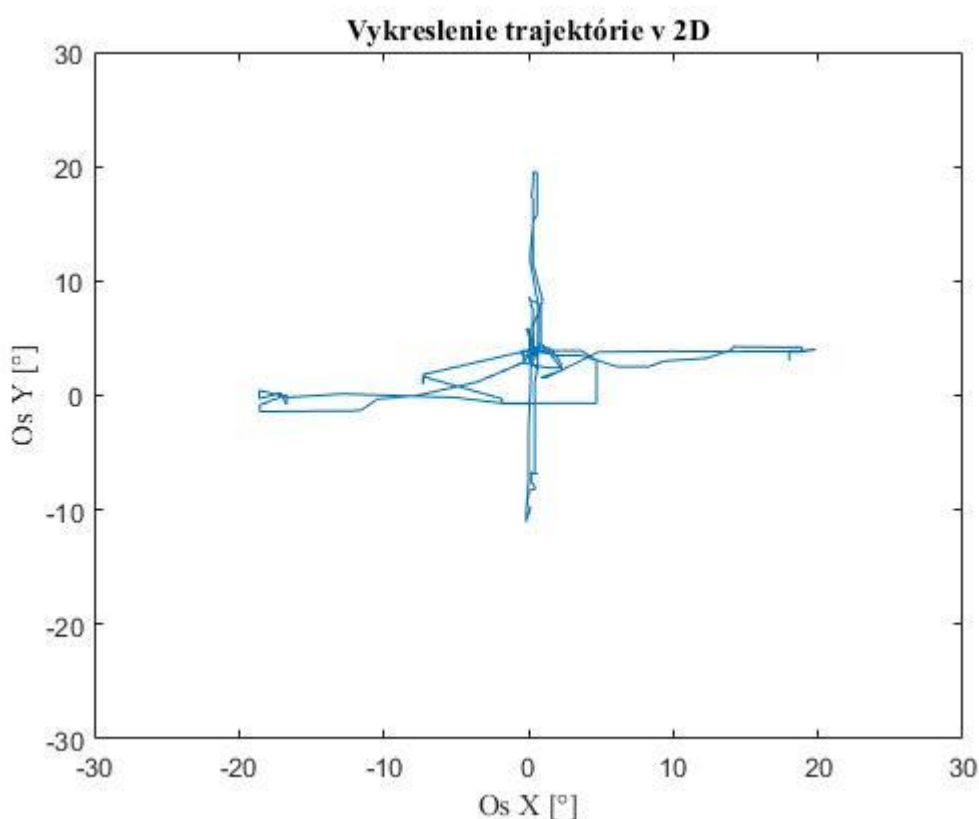
Vykreslenie trajektórie v 3D



Obrázok 6.2.1 Vykreslenie trajektórie pohybu v 3D

Vykreslenie trajektórie v 3D predstavuje vykreslenie polohy probanda počas celého merania, teda nie len počas plnenia prvej časti, z ktorej bol hodnotený rozsah pohybu. Toto vykreslenie umožňuje užívateľovi graf natáčať do ľubovoľných uhlov a teda má možnosť sledovať polohu probanda z rôznych uhlov pohľadu a vizuálne porovnávať dosiahnutý rozsah na jednotlivých osiach.

Druhá časť skriptu predstavuje vykreslenie trajektórie na osiach X a Y počas náklonu do strán (Obrázok 6.2.2).



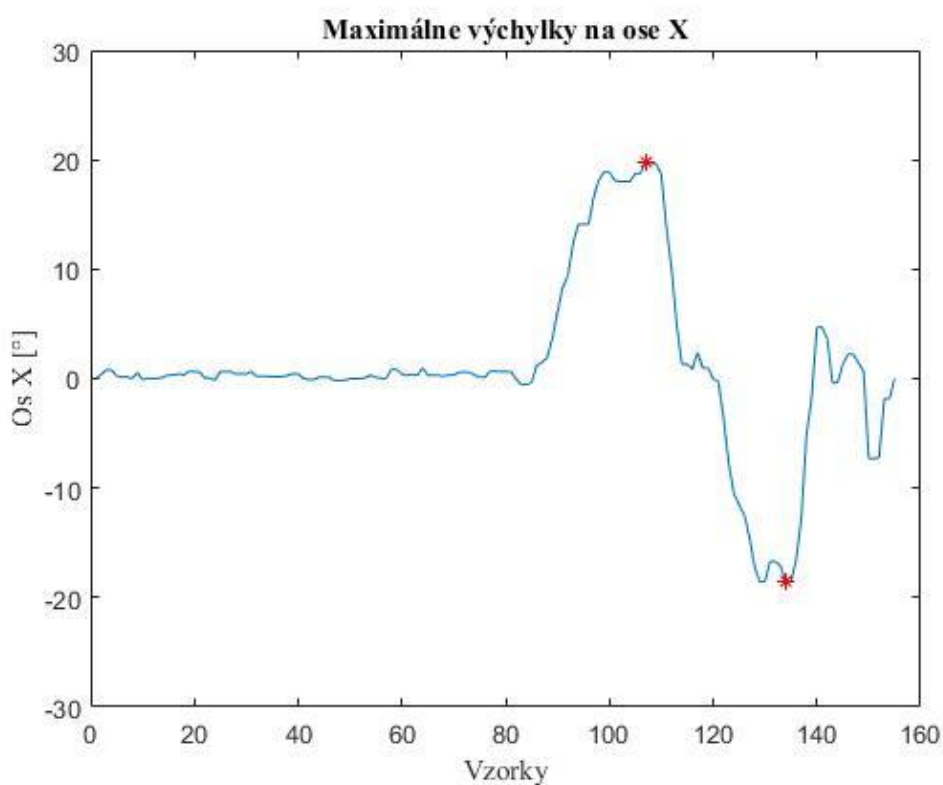
Obrázok 6.2.2 Vykreslenie trajektórie pohybu v 2D

Vykreslenie trajektórie pohybu v 2D užívateľovi umožňuje vizuálne posúdiť náklon probanda do strán. Z obrázku 6.2.2 je patrné, že vybraný proband nedosiahol rovnaký náklon do všetkých strán. Náklon vpravo a vľavo (os X) sa zdá byť približne rovnaký do oboch strán. Náklon vpred a vzad (os Y) sa líši, pričom náklon vpred robil probandovi menší problém ako náklon vzad, pretože pri náklone vpred dosiahol väčší rozsah pohybu.

Takéto vykreslenie trajektórie pohybu umožňuje užívateľovi posúdiť aj **symetriu pohybu**, teda porovnanie jednotlivých protipohybov. Na obrázku 6.2.2 môžeme vidieť, že sa proband pri náklone na osi X, teda vpavo a vľavo nepohyboval rovnomerne, ale na každú stranu vykonal náklon z iného miesta, čím spôsobil, že trajektória na osi X nie je lineárna. O trajektórii na Y – osi môžeme povedať, že je lineárna, pretože sa javí ako rovná čiara spojená náklonom vpred a vzad.

Tretia časť skriptu umožňuje vykreslenie signálov z jednotlivých osí gyroskopu a nájdenie maximálnych výchyliek na jednotlivých osiach.

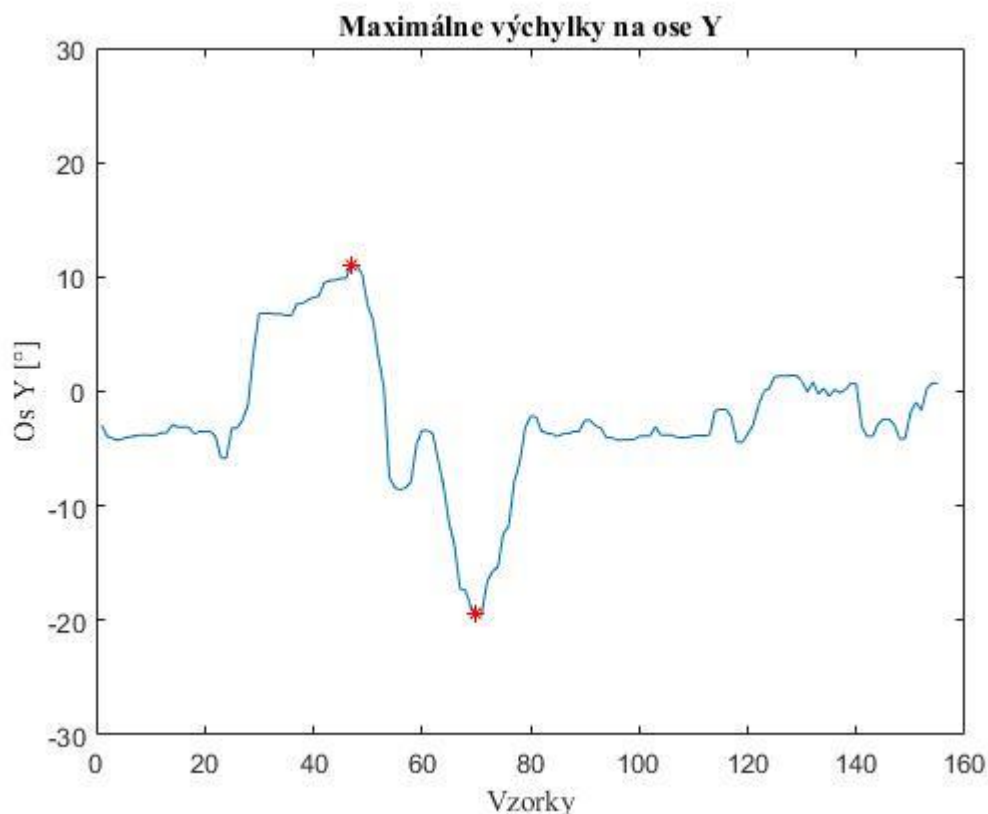
Algoritmus na vyhľadávanie **maximálnych výchyliek** pozostáva z usporiadania vektoru **GxR**, **GyR** a **GzR** od najväčšej hodnoty po najmenšiu a naopak, pričom sú získané konkrétne hodnoty a indexy bodov vo vektore. Vybráním prvého prvku z usporiadaných vektorov sú získané body, ktoré odpovedajú **najväčšiemu náklonu** vpravo a vľavo, vpred a vzad, hore a dolu. Tieto body boli následne vykreslené do grafu spolu so signálom z danej osi. Jednotlivé hodnoty najväčšieho náklonu boli vypísané do príkazového riadku.



Obrázok 6.2.3 Signál z osi X s vyznačenými maximálnymi hodnotami

Obrázok 6.2.3 predstavuje vykreslený signál z osi X, na ktorom sú vyznačené maximálne hodnoty náklonu do jednotlivých smerov – vpravo a vľavo.

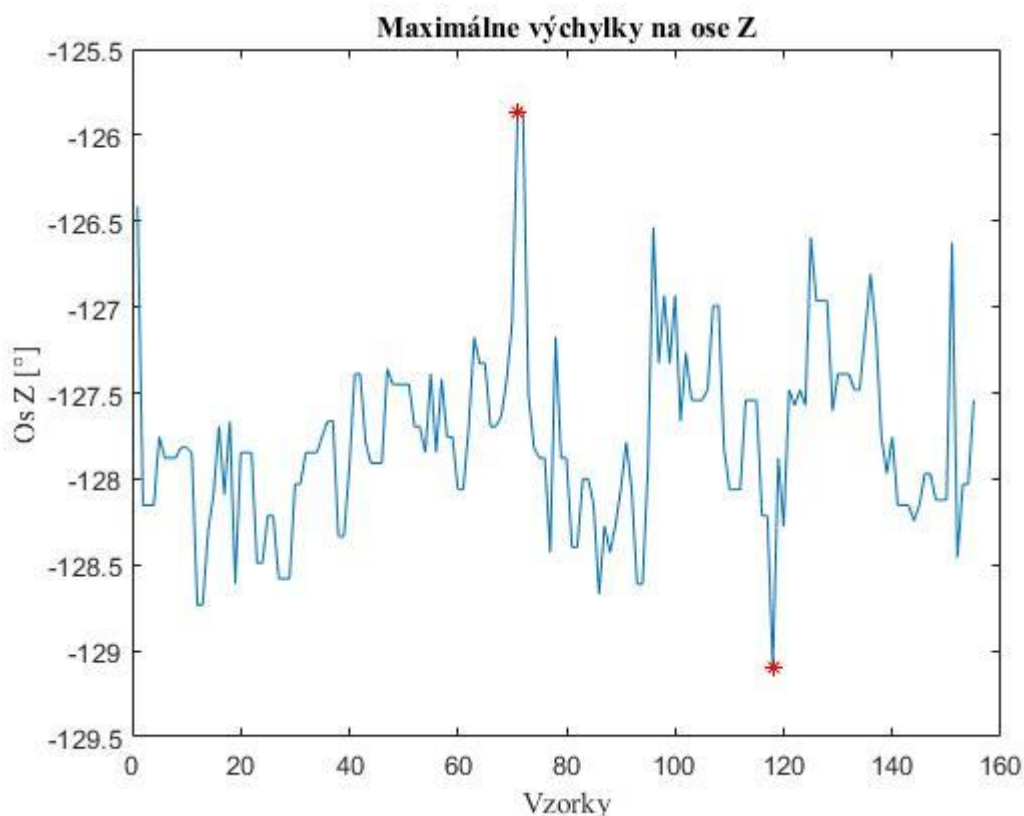
Najvyššia dosiahnutá hodnota pri náklone vpravo bola $19,8^\circ$. Najvyššia dosiahnutá hodnota pri náklone vľavo bola $18,6^\circ$. Rozdiel medzi týmito hodnotami predstavoval $38,4^\circ$.



Obrázok 6.2.4 Signál z osi Y s vyznačenými maximálnymi hodnotami

Obrázok 6.2.4 predstavuje vykreslený signál z osi Y, na ktorom sú vyznačené maximálne hodnoty náklonu do jednotlivých smerov – vpred a vzad.

Najvyššia dosiahnutá hodnota pri náklone vpred bola $19,5^\circ$. Najvyššia dosiahnutá hodnota pri náklone vzad bola 11° . Rozdiel medzi týmito hodnotami predstavoval $30,5^\circ$. Zo získaného rozdielu je patrné, že náklon vpred a vzad bol rádovo približne rovnaký ako náklon vpravo a vľavo.

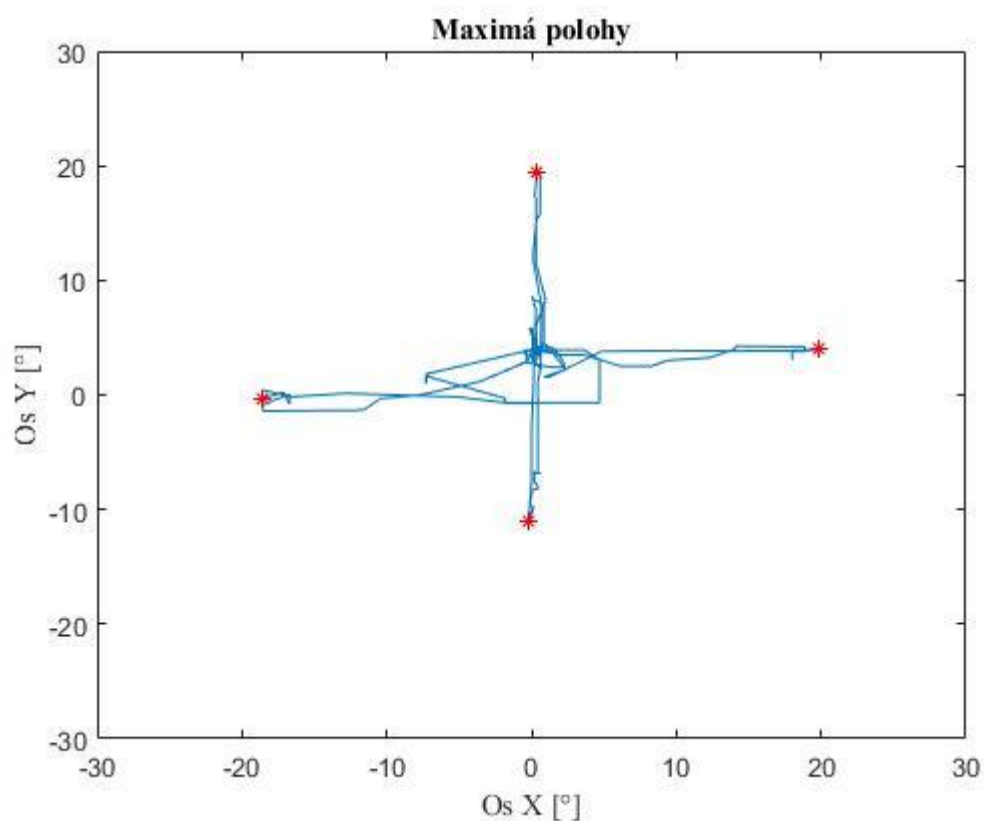


Obrázok 6.2.5 Signál z osi Z s vyznačenými maximálnymi hodnotami

Obrázok 6.2.4 predstavuje vykreslený signál z osi Z, na ktorom sú vyznačené maximálne hodnoty náklonu do jednotlivých smerov – hore a dolu.

Najvyššia dosiahnutá hodnota pri náklone smerom hore bola $129,1^{\circ}$. Najvyššia dosiahnutá hodnota pri náklone smerom dolu bola $125,8^{\circ}$. Rozdiel medzi týmito hodnotami predstavoval $3,3^{\circ}$. Z rozdielu medzi získanými hodnotami je patrné, že náklon smerom hore a dolu je zanedbateľný v porovnaní s náklonom vpred, vzad, vľavo a vpravo.

Štvrtá časť skriptu sa zaoberá vykresľovaním maximálnych polôh na trajektórii pohybu. Súradnice vykresľovaných bodov boli získané z už spomínaného vyhľadávania najvyššej, respektíve najnižšej hodnoty usporiadaného vektora z jednotlivých osí. Týmto hodnotám bol zároveň priradený index polohy vo vektore. Kombináciou hodnôt a indexov polôh boli získané súradnice bodov, ktoré boli následne vkreslené do grafu znázorňujúceho rozsah pohybu probanda, čo môžeme vidieť na obrázku 6.2.6.



Obrázok 6.2.6 Rozsah pohybu probanda s vyznačenými maximami polohy

Piata časť skriptu sa zaoberá stanovením dosiahnutého percentuálneho rozsahu pohybu v porovnaní s plným rozsahom prístroja. Plný rozsah prístroja nájdeme v tabuľke 6.1. Z tabuľky je patrné, že meraním pomocou prototypu snímača polohy nebolo možné získať symetrické hodnoty náklonu do strán, čo bolo spôsobené umiestnením snímača na pravej nožičke dynamického vertikalizátora.

Tabuľka 6.1 Plné rozsahy dynamického vertikalizátora

Smer	Hodnota
Vpred	64,12°
Vzad	43,18°
Rozdiel hodnôt	107,3°
Vpravo	48,97°
Vľavo	57,19°
Rozdiel hodnôt	106,16°

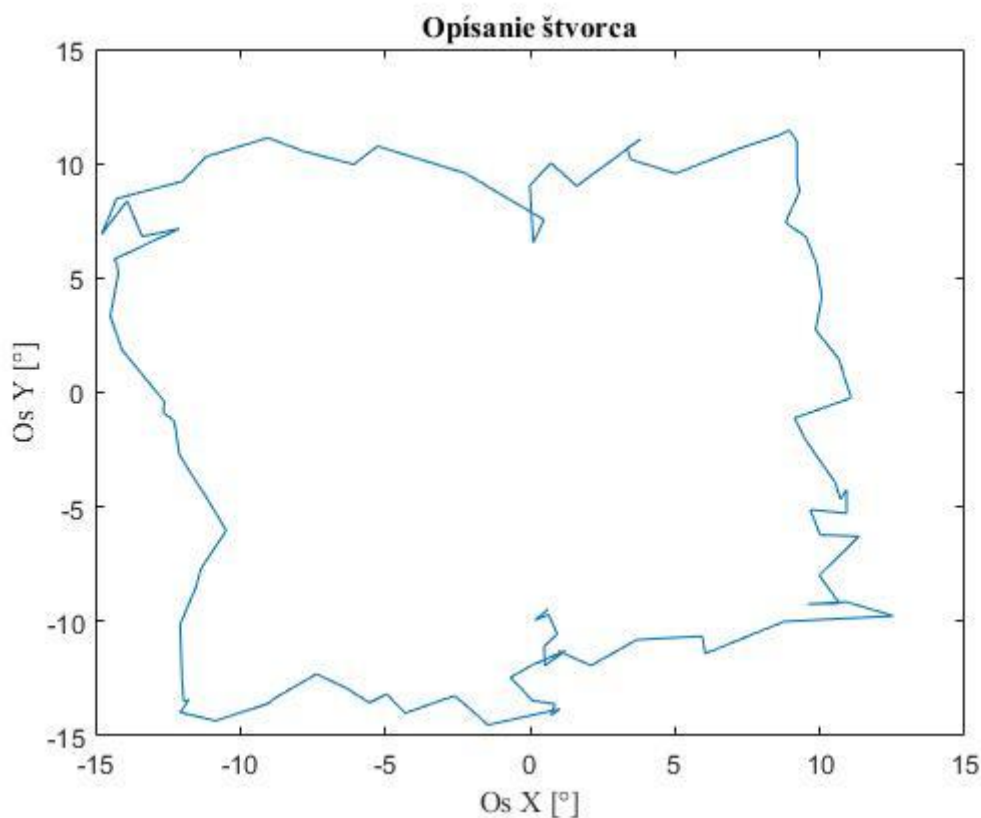
Prepočítavanie dosiahnutého rozsahu na percentá sa uskutočňovalo podľa vzorca (6).

$$\text{percentá} = \frac{\text{maximálny dosiahnutý rozsah} \cdot 100}{\text{maximálny rozsah prístroja}} [\%], \quad (6)$$

U vybraného probanda bol percentuálny rozsah náklonu vpravo 38 %, náklonu vľavo 34,7 %, náklonu vpred 17,1 % a náklonu vzad 45,2 % z plného rozsahu prístroja.

6.2.2 Zrýchlenie a rýchlosť pohybu

Zrýchlenie a rýchlosť pohybu probanda boli hodnotené z dát získaných pri opisovaní štvorca. Príklad opísaného štvorca môžeme vidieť na obrázku 6.2.7.



Obrázok 6.2.7 Trajektória pohybu pri opisovaní štvorca pohybom tela

Porovnávaná bola priemerná rýchlosť a najväčšie zrýchlenie dosiahnuté na protichodných stranách štvorca, teda priemerná rýchlosť a najväčšie zrýchlenie dosiahnuté na osi X a Y počas vykonávania pohybu.

Priemerná rýchlosť pohybu bola prepočítaná z vektorov zrýchlenia $\mathbf{AxV}$ a $\mathbf{AyV}$. Keďže zrýchlenie je deriváciou rýchlosti podľa času, na prepočet bolo potrebné vytvoriť vektor $\mathbf{t}$, ktorý reprezentoval časovú základňu. Prepočet rýchlosti sa uskutočnil pomocou príkazu `cumtrapz(t,AxV)`, prípadne `cumtrapz(t,AyV)`.

Priemerná dosiahnutá rýchlosť bola následne prepočítaná pomocou príkazu `median`, ktorý vektor spriemeroval a jeho výstupom bola priemerná dosiahnutá rýchlosť na danej osi.

Hodnota najväčšieho zrýchlenia bola získaná zoradením vektora $\mathbf{AxV}$ a $\mathbf{AyV}$ od najvyššej hodnoty po najnižšiu, pričom najvyššia hodnota predstavovala najvyššie dosiahnuté zrýchlenie na danej osi.

Priemerná dosiahnutá rýchlosť vybraného probanda na vrchnej a spodnej strane štvorca, teda na ose X, bola 0,003 m/s. Najvyššie zrýchlenie bolo 0,06 m/s². Na osi Y, teda na pravej a ľavej strane štvorca bola priemerná dosiahnutá rýchlosť 0,05 m/s a najvyššie zrýchlenie 0,1 m/s².

Z trajektórie vytvorenej pri opisovaní štvorca môžeme vizuálne hodnotiť dokonalosť opísania útvaru. Opísaný štvorec by mal byť čo najsymetrickejší, teda vrchná, spodná, pravá a ľavá strana by mali byť rovnako dlhé a navzájom kolmé.

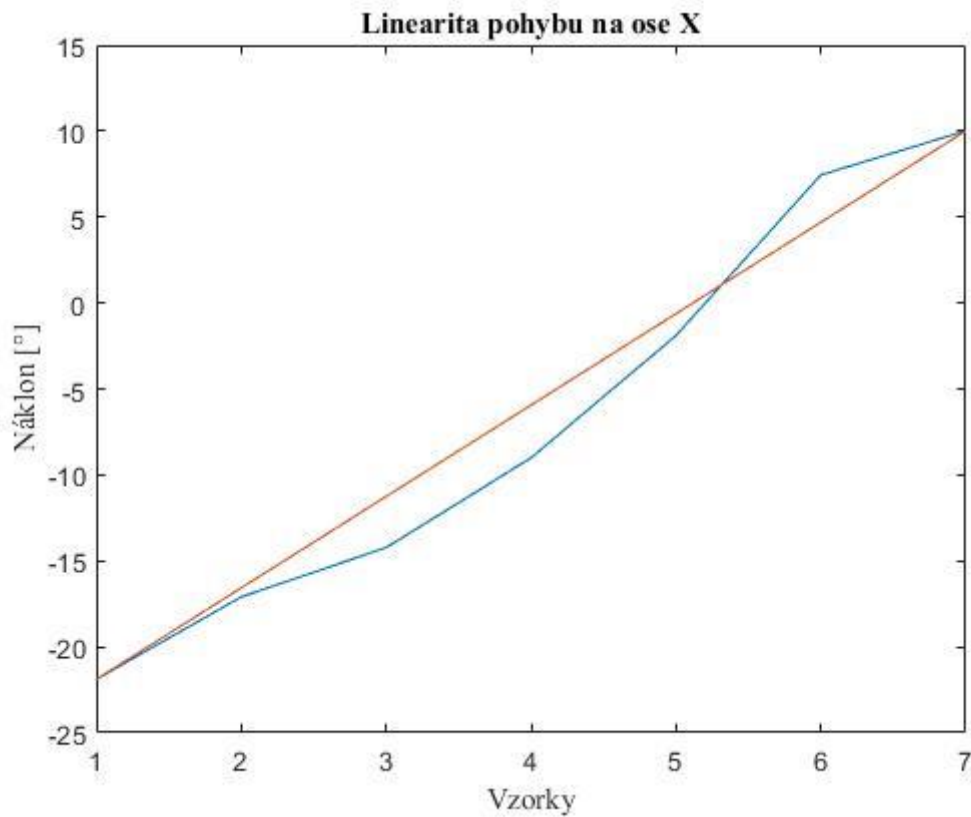
6.2.3 Linearita pohybu

Linearita pohybu bola hodnotená z dát získaných pri opísaní čo najrovnejšej čiary z jednej strany na druhú. Vektor potrebný na toto hodnotenie bol vektor $\mathbf{GyL}$, ktorý predstavoval pohybu na osi X (obrázok 6.2.8).

Algoritmus na hodnotenie linearity pohybu pozostáva z porovnania dosiahnutého pohybu s ideálnym stavom, teda stavom, kedy by pohyb predstavoval rovnú čiaru. Tento ideálny pohyb by bol pomocou vektora $\mathbf{GyL}$ vykreslený ako rovná priamka z bodu, kde pohyb začal do bodu, kde pohyb skončil. Vykreslenie tohto pomocného ideálneho vektora môžeme vidieť na obrázku 6.2.8 červenou farbou. Modrou farbou je vykreslený vektor $\mathbf{GyL}$, ktorý je nameraný počas pohybu probanda. Na zistenie percentuálneho vychýlenia

od požadovanej trajektórie bol použitý cyklus pre výpočet Euklidovskej vzdialenosti podľa vzorca (7), kde premenná **vzdialenosť** predstavovala vektor hodnôt Euklidovských vzdialeností vektora **GyL** od vektora **pom**. Následne bola z vektora **vzdialenosť** vypočítaná priemerná hodnota, ktorá bola použitá na percentuálne vyjadrenie linearity pohybu.

$$vzdialenosť = \sqrt{(GyL(i) - pom(i))^2 - (GyL(j) - pom(j))^2}, \quad (7)$$



Obrázok 6.2.8 Linearita pohybu na osi X

Na zistenie percentuálneho dosiahnutia linearity pohybu bol následne použitý vzorec (8), kde premenná *vektor* predstavovala dĺžku vektora **GyL** a premenná *Euklid* predstavovala vypočítanú priemernú Euklidovskú vzdialenosť.

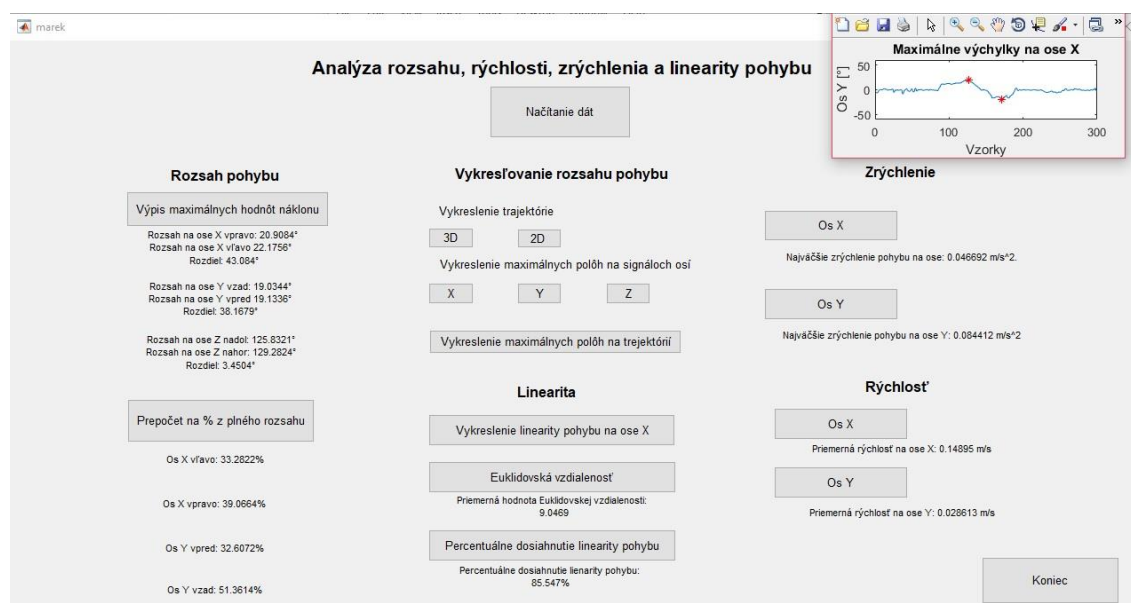
$$percentá = 100 - \frac{Euklid * 100}{vektor} [\%], \quad (8)$$

Za dobrý výsledok bola považovaná čo najväčšia dosiahnutá hodnota. Proband, ktorého vykreslený vektor môžeme vidieť na obrázku 6.2.8 dosiahol vychýlenie 96 %, čo môžeme považovať za veľmi dobrý výsledok.

6.3 Grafické prostredie

Grafické prostredie slúži na analýzu nameraných dát používateľom. Prostredie bolo vytvorené v programe Matlab R2017a pomocou funkcie *guide*, ktorá umožňuje realizáciu grafického prostredia.

Vytvorené grafické prostredie sa skladá z troch častí – analýzy rozsahu pohybu, analýzy rýchlosti a zrýchlenia pohybu a analýzy linearity pohybu (obr. 6.3.1 – detailnejší obrázok sa nachádza v prílohách práce). Program je spúšťaný pomocou príkazu *marek* v príkazovom riadku. Použité algoritmy sú upravené algoritmy skriptu *spracovanie.m* pre potreby grafického prostredia.



Obrázok 6.3.1 Analýza rozsahu pohybu v grafickom prostredí

Prvá časť analýzy slúži na načítanie dát vybraného probanda do programu. Po stlačení tlačítka **Načítanie dát** sa na obrazovke objaví okno, cez ktoré si užívateľ vyberie požadované dáta.

Časť analýzy rozsahu pohybu slúži na vykreslenie nameraných a spracovaných dát. Tlačítka **3D** a **2D** vykresľujú 3D a 2D trajektóriu pohybu vybraného probanda.

Tlačítka **X**, **Y** a **Z** vykresľujú signály z konkrétnych osí, pričom na každom zo signálov je vykreslená najvyššia a najnižšia hodnota, čo predstavuje najvyšší náklon do strán. Príklad vykreslenia môžeme vidieť na obrázku 6.3.1 vpravo hore. Tlačítko **Maximá polohy** vykresľuje maximálne hodnoty náklonu do strán na 2D trajektórii pohybu.

Ďalšia časť analýzy rozsahu pohybu sa zaoberá výpisom maximálnych hodnôt náklonu do všetkých strán pomocou tlačítka **Výpis maximálnych hodnôt náklonu** a prepočtom a následným výpisom týchto hodnôt na percentuálne dosiahnutie náklonu v porovnaní s plným rozsahom prístroja pomocou tlačítka **Prepočet na % z plného rozsahu**.

Druhá časť programu slúži na analýzu rýchlosti, zrýchlenia a linearity pohybu. V prvej časti tejto analýzy, ktorá je venovaná zrýchleniu pohybu, je prostredníctvom tlačítka **Os Y** a **Os X** možné vykresliť priebeh zrýchlenia na osiach a zároveň program vypíše najväčšie dosiahnuté zrýchlenie na danej osi.

V časti Rýchlosť umožňuje tlačítko **Os X** a **Os Y** vykreslenie priebehu rýchlosti na osiach a program vypíše priemernú rýchlosť pohybu na danej osi.

Posledná časť programu sa zaoberá linearitou pohybu probanda. Vykreslenie signálu na osi, na ktorej proband opisoval čo najrovnejšiu čiaru, teda na osi X je možné prostredníctvom tlačítka **Vykreslenie linearity pohybu na osi X**. Tlačítko **Euklidovská vzdialenosť** slúži na vypísanie priemernej Euklidovskej vzdialenosti, ktorá slúžila na prepočet linearity na percentá. Po stlačení tlačítka **Percentuálne dosiahnutie linearity pohybu** program vypíše hodnotu percentuálneho dosiahnutia linearity pohybu.

Tlačítko **Koniec** umožňuje ukončenie programu a zatvorenie všetkých programových okien.

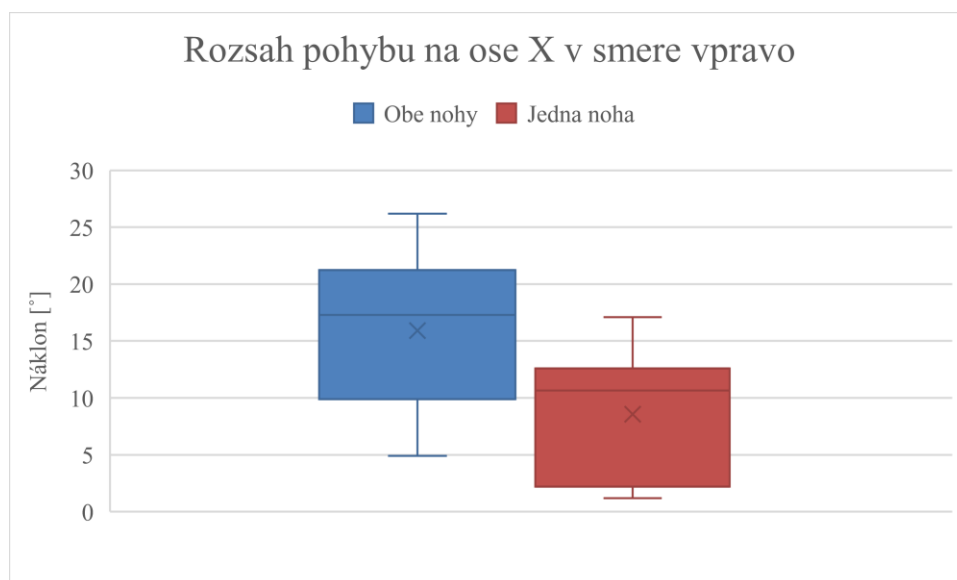
7 ŠTATISTICKÉ VYHODNOTENIE

Cieľom štatistického vyhodnotenia dát je zistenie rozdielov pri prevedení pohybov na oboch a na jednej nohe. Štatistické spracovanie dát sa odohráva v programe Excel a použitá metóda bola Studentovo rozdelenie, konkrétne dvojvýberový t-test s rovnosťou rozptylov, prípadne bez rovnosti rozptylov.

7.1 Vyhodnotenie rozsahu pohybu

Na vyhodnotenie rozsahu pohybu boli použité dáta získané z analýzy rozsahu pohybu probandov, konkrétne náklon vpred, vzad, vpravo, vľavo a percentuálne dosiahnutie náklonu v porovnaní s plným rozsahom prístroja do všetkých strán.

Hypotéza, ktorá mala byť potvrdená alebo zamietnutá v prvom prípade, teda pri náklone do strán, bola: Proband dosiahol väčší náklon pri státi na oboch nohách ako pri státi na jednej nohe v danom smere. Táto hypotéza bola prijatá len v prípade, kedy proband vykonával náklon smerom vpravo. V ostatných prípadoch nebol štatistický rozdiel významný. Na obrázku 6.1.1 môžeme vidieť krabicový graf rozdelenia dát pri státi na oboch nohách a na jednej nohe pri náklone vpravo.



Obrázok 7.1.1 Rozdelenie dát pri náklone vpravo

V tabuľke 7.1 nájdeme konkrétne výsledky t-testu.

Tabuľka 7.1 Výsledky t-testu pri náklone vpravo

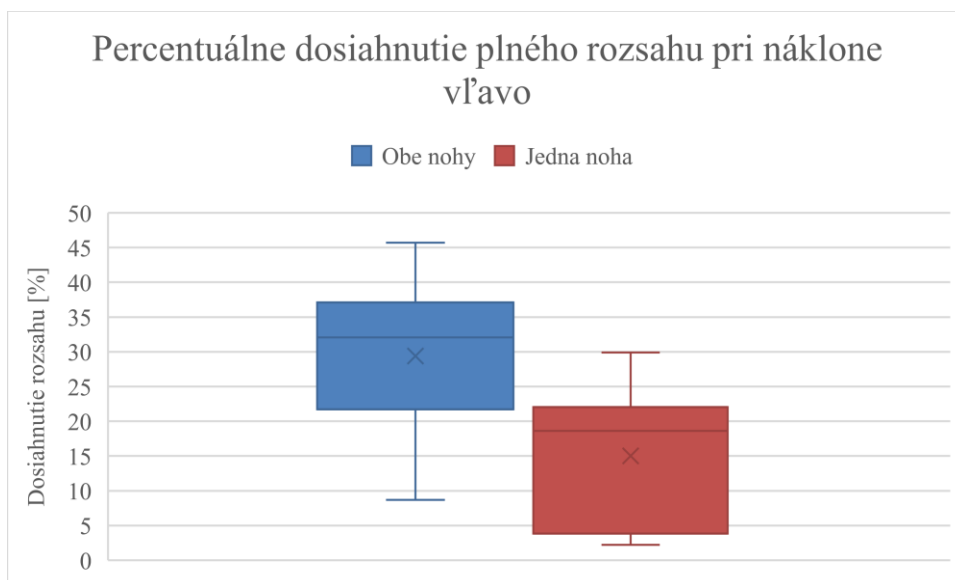
	Obe nohy	Jedna noha
Priemer	15,92°	8,57°
Rozptyl	46,35°	33,68°
p-hodnota	0,009	

Z výsledkov t-testu je zrejmé, že p-hodnota je menšia ako 0,05, z čoho vyplýva, že hypotéza je prijatá. Rozdiel vo veľkosti náklonu pri státí na oboch nohách bol štatisticky významnejší, než rozdiel pri státí na jednej nohe pri náklone vpravo. Výsledky testu pri náklone do ostatných strán nájdeme v tabuľke 7.2.

Tabuľka 7.2 Výsledky t-testu pri náklonoch

	Náklon vľavo		Náklon vpred		Náklon vzad	
	Obe nohy	Jedna noha	Obe nohy	Jedna noha	Obe nohy	Jedna noha
Priemer	16,81°	13,12°	14,81°	15,28°	15,53°	15,44°
Rozptyl	36,99°	21,17°	34,75°	33,56°	28,33°	28,32°
p-hodnota	0,07		0,43		0,48	
Hypotéza	zamietnutá		zamietnutá		zamietnutá	

Hypotéza pri percentuálnom dosiahnutí náklonu v porovnaní s plným rozsahom prístroja znela: Percentuálne dosiahnutie náklonu v porovnaní s plným rozsahom prístroja bude vyššie pri státí na oboch nohách, než pri státí na jednej nohe v danom smere. Táto hypotéza bola prijatá len pri náklone vľavo. V ostatných prípadoch nebol štatistický rozdiel významný. Na obrázku 7.1.2 môžeme vidieť krabicový graf rozdelenia dát pri státí na oboch nohách a na jednej nohe pri náklone vľavo.



Obrázok 7.1.2 Percentuálne dosiahnutie plného rozsahu pri náklone vľavo

V tabuľke 7.3 nájdeme konkrétne výsledky t-testu.

Tabuľka 7.3 Výsledky t-testu percentuálneho dosiahnutia plného rozsahu pri náklone vľavo

	Obe nohy	Jedna noha
Priemer	29,38%	15%
Rozptyl	134,54%	102,94%
p-hodnota	0,004	

Z výsledkov t-testu je zrejmé, že p-hodnota je menšia ako 0,05, z čoho vyplýva, že hypotéza je prijatá. Rozdiel v percentuálnom dosiahnutí náklonu v porovnaní s plným rozsahom prístroja pri státi na oboch nohách bol štatisticky významnejší, než rozdiel pri státi na jednej nohe pri náklone vľavo. Výsledky testu pri náklone do ostatných strán nájdeme v tabuľke 7.4.

Tabuľka 7.4 Výsledky t-testu percentálneho dosiahnutia plného rozsahu

	Dosah vpravo		Dosah vpred		Dosah vzad	
	Obe nohy	Jedna noha	Obe nohy	Jedna noha	Obe nohy	Jedna noha
Priemer	34,05%	26,76%	22,91%	24,08%	34,7%	35,43%
Rozptyl	151,79%	88,28%	68,42%	69,08%	198,96%	177,89%

p-hodnota	0,08	0,38	0,45
Hypotéza	zamietnutá	zamietnutá	zamietnutá

7.2 Vyhodnotenie zrýchlenia a rýchlosti pohybu

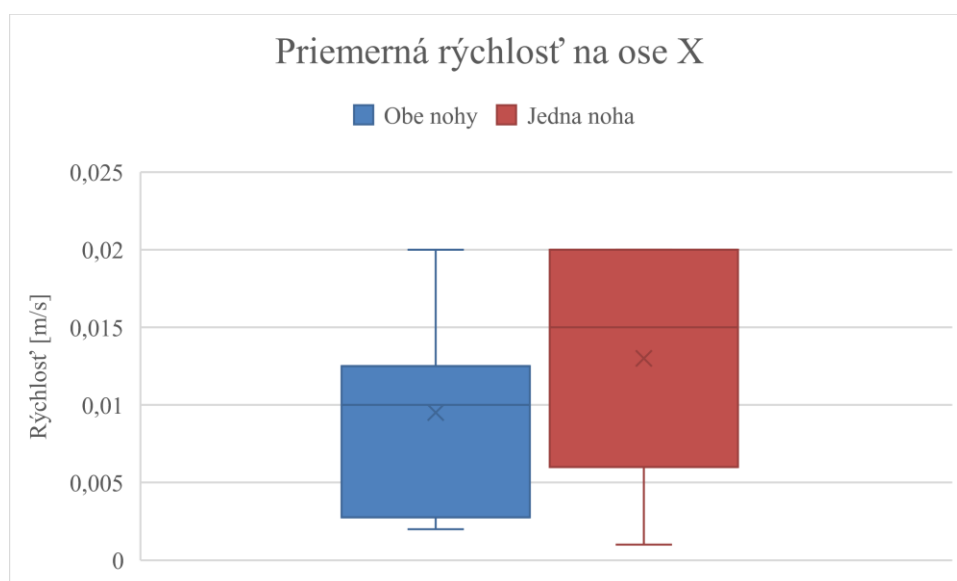
Na vyhodnotenie priemernej rýchlosti a najväčšieho zrýchlenia na osiach boli použité dáta získané z analýzy pohybu pri opisovaní štvorca. Z analýzy bola získaná priemerná rýchlosť na osi X a Y a najväčšie zrýchlenie na týchto osiach.

Hypotéza pri štatistickom hodnotení priemernej rýchlosti znela: Proband stojaci na oboch nohách dosiahol vyššiu priemernú rýchlosť na danej ose, než proband stojaci na jednej nohe. Táto hypotéza nebola potvrdená v žiadnom prípade. Na obrázku 7.2.1 môžeme vidieť krabicový graf rozdelenia dát priemernej rýchlosti na ose X pri státi na oboch a na jednej nohe.

V tabuľke 7.5 nájdeme konkrétne výsledky t-testu.

Tabuľka 7.5 Výsledky t-testu priemernej rýchlosti na ose X

	Obe nohy	Jedna noha
Priemerná hodnota	0,01 m/s	0,01 m/s
Rozptyl	$4,21 \cdot 10^{-5}$ m/s	$6,11 \cdot 10^{-5}$ m/s
p-hodnota	0,12	



Obrázok 7.2.1 Priemerná rýchlosť na ose X

Z výsledkov t-testu je zrejmé, že p-hodnota je väčšia ako 0,05, z čoho vyplýva, že hypotéza je zamietnutá. Rozdiel v dosiahnutej priemernej rýchlosti na osi X pri státí na oboch nohách nebol štatisticky významnejší, než rozdiel pri státí na jednej nohe. Výsledky testu priemerných rýchlostí na osi Y nájdeme v tabuľke 6.6.

Tabuľka 7.6 Výsledky t-testu priemernej rýchlosti na osi Y

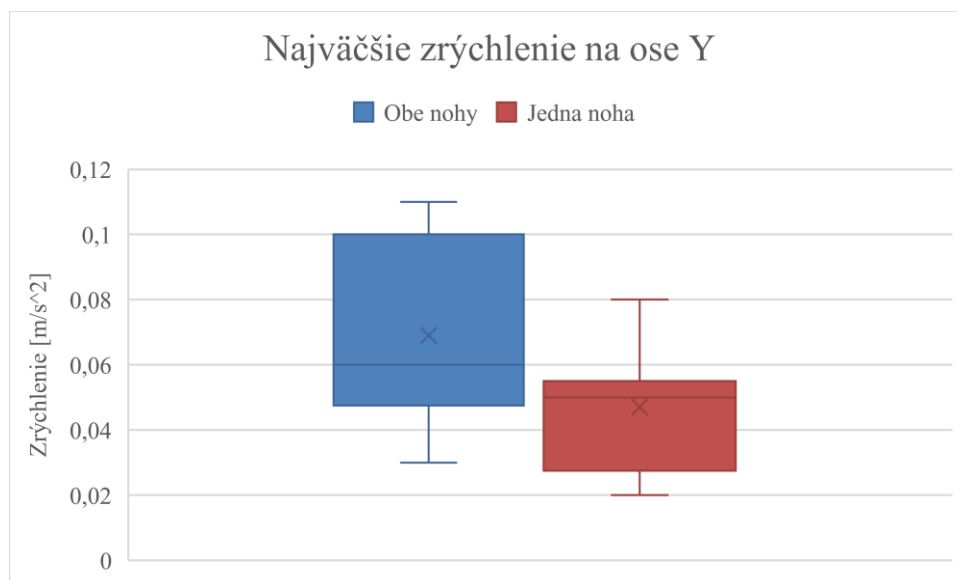
	Obe nohy	Jedna noha
Priemer	0,02 m/s	0,02 m/s
Rozptyl	$2,2 \cdot 10^{-4}$ m/s	$1,72 \cdot 10^{-4}$ m/s
p-hodnota	0,49	
Hypotéza	zamietnutá	

Hypotéza pri štatistickom hodnotení najväčšieho dosiahnutého zrýchlenia znela: Proband stojaci na oboch nohách dosiahol väčšie zrýchlenie na danej osi, než proband stojaci na jednej nohe. Táto hypotéza bola potvrdená v prípade, kedy proband vykonával pohyb pozdĺž osi Y. V druhom prípade bola táto hypotéza zamietnutá. Na obrázku 7.2.2 môžeme vidieť krabicový graf rozdelenia dát najväčšieho dosiahnutého zrýchlenia na osi Y pri státí na oboch a na jednej nohe.

V tabuľke 7.7 nájdeme konkrétne výsledky t-testu.

Tabuľka 7.7 Výsledky t-testu najväčšieho dosiahnutého zrýchlenia na osi Y

	Obe nohy	Jedna noha
Priemer	$0,07 \text{ m/s}^2$	$0,05 \text{ m/s}^2$
Rozptyl	$0,0008 \text{ m/s}^2$	$0,0003 \text{ m/s}^2$
p-hodnota	0,03	



Obrázok 7.2.2 Najväčšie zrýchlenie na ose Y

Z výsledkov t-testu je zrejmé, že p-hodnota je menšia ako 0,05, z čoho vyplýva, že hypotéza je prijatá. Rozdiel v dosiahnutom najväčšom zrýchlení na osi Y pri státí na oboch nohách bol štatisticky významnejší, než pri státí na jednej nohe. Výsledky testu najväčšieho zrýchlenia na osi X nájdeme v tabuľke 7.8.

Tabuľka 7.8 Výsledky t-testu najväčšieho dosiahnutého zrýchlenia na osi X

	Obe nohy	Jedna noha
Priemer	0,042 m/s ²	0,048 m/s ²
Rozptyl	0,0009 m/s ²	0,0005 m/s ²
p-hodnota	0,32	
Hypotéza	zamietnutá	

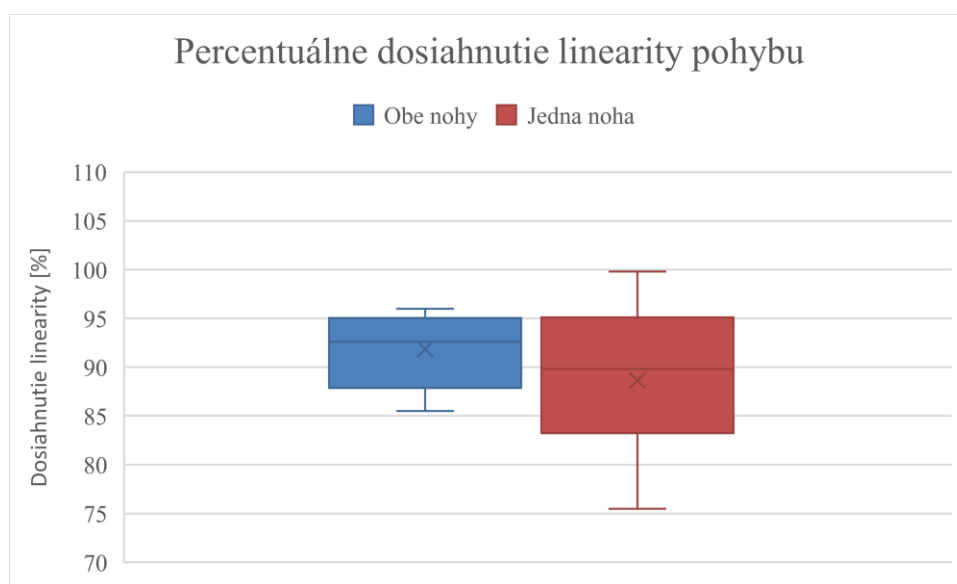
7.3 Vyhodnotenie linearity pohybu

Na štatistické vyhodnotenie linearity pohybu boli použité dáta získané z analýzy linearity pohybu. Hypotéza znela: Percentuálne dosiahnutie linearity pohybu pri státí na oboch nohách bude vyššie, než pri státí na jednej nohe. Táto hypotéza však potvrdená nebola. Na obrázku 7.3.1 môžeme vidieť krabicový graf rozdelenia dát percentuálneho dosiahnutia linearity pohybu pri státí na oboch a na jednej nohe.

V tabuľke 7.9 nájdeme konkrétne výsledky t-testu.

Tabuľka 7.9 Výsledky t-testu percentuálneho dosiahnutia linearitu pohybu

	Obe nohy	Jedna noha
Priemer	91,85%	88,63%
Rozptyl	13,55%	66,42%
p-hodnota	0,14	



Obrázok 7.3.1 Percentuálne dosiahnutie linearitu pohybu

Z výsledkov t-testu je zrejmé, že p-hodnota je väčšia ako 0,05, z čoho vyplýva, že hypotéza je zamietnutá. Rozdiel v percentuálnom dosiahnutí linearitu pohybu pri státi na oboch nohách nebol štatisticky významnejší, než pri státi na jednej nohe.

8 HODNOTENIE PROTOTYPU SNÍMAČA POLOHY A VÝSLEDKOV

Cieľom tejto kapitoly je zhodnotiť navrhnutý a skonštruovaný prototyp snímača polohy a dosiahnuté výsledky.

8.1 Prototyp snímača polohy

Navrhnutý a skonštruovaný prototyp snímača polohy bol konštrukčne v poriadku a pre daný účel plne vyhovoval.

Problém miestami spôsoboval počítač, ktorý nestíhal spracovávať a preposielať dáta medzi jednotlivými spustenými programami, čo spôsobovalo občasné výpadky pri ovládaní hry.

Problémy by pri ďalšom použití snímača mohli spôsobiť zvolené rozsahy, ktoré by mohli byť nedostačujúce v prípade použitia iného dynamického vertikalizátora. Tento problém sa dá vyriešiť preprogramovaním senzora.

8.2 Hodnotenie výsledkov

Výsledky získané z analýzy dát a štatistického spracovania môžeme vidieť v predošlých kapitolách.

Podľa predpokladov boli očakávané väčšie rozdiely pri pohyboch vykonávaných pri státí na jednej a na oboch nohách, čo sa však nepotvrdilo. Teoreticky by mal mať človek väčšiu stabilitu pri státí na oboch nohách a tým pádom aj pohyby vykonávané na oboch nohách by mali byť výraznejšie. Rozdiel medzi státím na oboch a na jednej nohe eliminuje samotný dynamický vertikalizátor, v ktorom boli probandi upevnení, a tak mali dostatočnú stabilitu aj pri státí na jednej nohe. Ak by bolo meranie vykonané mimo prístroja, výsledky by pravdepodobne dopadli podľa predpokladov, teda rozdiel medzi státím na oboch a na jednej nohe by bol omnoho výraznejší.

Faktor, ktorý skresľoval výsledky pri státí na jednej nohe bol fakt, že dynamický vertikalizátor kládol príliš veľký odpor pri náklone do strany, na ktorej proband stál – v prípade, že proband stál na pravej nohe, dynamický vertikalizátor prakticky znemožňoval probandovi náklon do pravej strany.

ZÁVER

Počas písania a plnenia úloh prvej časti bakalárskej práce som sa oboznámila s problematikou terapeutických hier a použitím dynamického vertikalizátora v procese rehabilitácie vo fyzioterapii. Cieľom bolo navrhnuť a skonštruovať polohový senzor, ktorý umožňoval zaznamenávanie dát a ovládanie počítačových hier prostredníctvom dynamického vertikalizátora.

Na návrh a konštrukciu polohového senzora bola použitá vývojová doska Arduino Leonardo Pro Micro a komponenty modul MPU-6050, ktorý predstavuje akcelerometer s gyroskopom a Bluetooth modul HC-05, z ktorých bol vytvorený prototyp snímača polohy umožňujúci bezdrôtový prenos údajov do počítača.

Programová časť zahŕňala prácu s programom Arduino IDE a Matlab R2017a, pričom v Arduino IDE bolo mojou úlohou naprogramovať prototyp snímača polohy tak, aby bolo možné prostredníctvom polohového senzora bezdrôtovo preposielať namerané pohybové dáta do počítača. V programe Matlab R2017a bol vytvorený skript, na základe ktorého bolo možné zo získaných nameraných dát ovládať počítačovú klávesnicu a dáta zo snímača polohy vykresľovať v reálnom čase a ukladať, čo umožnilo ich ďalšie spracovanie.

Po skonštruovaní hardwarovej časti prototypu snímača a naprogramovaní softwarovej časti sa uskutočnilo meranie polohových dát na desiatich dobrovoľníkoch. Prvé meranie spočívalo v hraní počítačovej hry pomocou snímača polohy upevneného na dynamickom vertikalizátore, v ktorom stáli jednotliví probandi. Z výsledkov tohto merania bolo zrejmé, že hranie hry bolo relatívne náročné, ale zábavné. Predmetom druhého merania bolo získanie dát na analýzu rozsahu, zrýchlenia, rýchlosti a linearitu pohybu probandov. Probandi plnili úlohy, z ktorých boli získané potrebné dáta. Toto meranie prebehlo v poriadku, bez výpadkov senzora a namerané dáta boli použiteľné k analýze.

Ďalšou časťou práce bola analýza získaných polohových dát z druhého merania, pri ktorej bol vytvorený algoritmus na spracovanie nameraných dát. Hodnotený bol rozsah pohybu do všetkých smerov, zrýchlenie a rýchlosť pohybu na jednotlivých osiach a linearita pohybu. Analýza prebiehala v programe vytvorenom pomocou GUI Matlabu. Z výsledkov analýzy bolo zrejmé, že každý proband používa individuálne pohybové

vzorke. Výsledky analýzy boli následne štatisticky spracované v programe Excel a vyhodnotené. Štatistické spracovanie potvrdilo očakávané predpoklady len v niekoľkých prípadoch a preukázalo sa, že pri pohybe vľavo dochádza k signifikantným rozdielom pri odrážaní sa jednou alebo oboma nohami, zatiaľ čo pre ostatné smery nebol signifikantný rozdiel pozorovaný.

Literatúra

- [1] NeuroRehabilitations 23. *Treadmill training with partial body weight support afterstroke: A review.* [online]. [cit. 2008]. Dostupné na URL: <<https://content.iospress.com/download/neurorehabilitation/nre00394?id=neurorehabilitation%2Fne00394>>
- [2] BIN, Emmanuelle. *La marche appareillée comme outil de rééducation chez patient paraplegique de niveau T10 ASIA B.* [online]. [cit. 2010 - 2011]. Dostupné na URL: <http://ifpek.centredoc.org/opac/doc_num.php?explnum_id=326>
- [3] RADTKA, HONE, BROWN, MASTICK, MELNICK, DOWLING. *Games Health J.* [online]. [cit. 2013-8]. Dostupné na URL: <<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3833678/>>
- [4] VODA, Zbyšek. *Průvodce světem Arduina.* Bučovice: Martin Stříž, 2015. ISBN isbn978-80-87106-90-7.
- [5] SMITH, Alan. *Introduction to Arduino: A piece of cake!.* CreateSpace Independent Publishing Platform, 2011, 170s. ISBN 978-1463698348.
- [6] LUBOŠ M. *Gyroskop a akcelerometr.* [online]. [cit. 2016-3-15]. Dostupné na URL: <<http://navody.arduino-shop.cz/navody-k-produktum/gyroskop-a-akcelerometr.html>>
- [7] NAVRÁTIL, Josef. *Arduino Nano a akcelerometr.* [online]. [cit. 2011-2-10]. Dostupné na URL: <http://www.josefnav.cz/Arduino_akcelerometr.html>
- [8] KREIDL, Marcel a Radislav ŠMÍD. *Technická diagnostika: senzory, metody, analýza signálu.* Praha: BEN - technická literatura, 2006. Senzory neelektrických veličin. ISBN 80-7300-158-6.
- [9] InvenSense Inc. *MPU-6000 and MPU-6050 Product Specification Revision 3.4.* [online]. [cit. 2013-8-19]. Dostupné na URL: <<https://www.invensense.com/wp-content/uploads/2015/02/MP>>
- [10] MAROTTO, SOLE, DESSI, MANCHINU, CARBONI, SERRA, *Orientation Analysis through a Gyroscope Sensor for Indoor Navigation Systems.* [online]. [cit. 2013-6]. Dostupné na URL: <https://www.thinkmind.org/index.php?view=article&articleid=sensordevices_2013_6_40_20221>

- [11] GME. *HC-05 Bluetooth Module*. [online]. [cit. 2017]. Dostupné na URL:<<https://www.gme.cz/data/attachments/dsh.772-148.1.pdf>>
- [12] ITCH. *Cube Dodger* [online]. [cit. 2017-5-2]. Dostupné na URL:<<https://ethan-the-drake.itch.io/cube-dodger>>

Zoznam symbolov, veličín a skratiek

AWR – Angular Random Walk

Arduino IDE – Arduino Integrated Development Environment

P/S prevodník - paralelne sériový prevodník

I/O – Input, Output

I2C – Inter-Integrated Circuit

SPI – Seriól Peripheral Interface

SCL – Synchronous Clock

SDA – Synchronous Data

DMP – Digital motion processor

USB – Universal Serial Bus

VCC – Voltage Collector Collector

GND – Ground

MEMS – Microelectromechanical Systems

RXD – Recieve Data

TXD – Transmitt Data

MISO – Master In, Slave Out

MOSI – Master Out, Slave In

Zoznam príloh

Príloha 1. Protokol merania 1

Príloha 2. Protokol merania 2

Príloha 3. Grafické prostredie programu

Príloha 4. Zdrojové kódy (CD)

Príloha 5. Štatistické spracovanie výsledkov (CD)

Príloha 6. Prototyp snímača polohy

PRÍLOHY

Príloha 1.

Protokol merania 1

ZAZNAMENANIE POHYBU SKUPINY DOBROVOĽNÍKOV POČAS HRANIA HRY NA DYNAMICKOM VERTIKALIZÁTORE

Pomôcky a programy:

Dynamický vertikalizátor, prototyp snímača polohy, Matlab R2017a, Cube Dodger

Popis merania:

Meranie sa uskutoční na ôsmich až desiatich náhodne vybraných dobrovoľníkoch. Cieľom merania je získať pohybové dáta z akcelerometra a gyroskopu, ktoré budú ďalej spracované v programe Matlab R2017a. Proces získania dát zahŕňa hranie hry Cube Dodger pomocou prototypu snímača polohy skonštruovaného vývojovou doskou Arduino Leonardo Pro Micro, akcelerometrom a gyroskopom MPU-6050 a Bluetooth modulom HC-05. Tento prototyp je umiestnený na nožičke dynamického vertikalizátora. Dáta budú do programu Matlab odosielané bezdrôtovo, budú uložené a pripravené na ďalšie spracovanie.

Postup merania:

1. Dobrovoľník sa postaví na plochu dynamického vertikalizátora určenú na státie. Obsluha zaistí kolieska, nastaví bedrovú podperu tak, aby tesne obopínala sedacie svaly dobrovoľníka, čím dobrovoľníka znehybní, prípadne nastaví kolenné podpery.
2. Obsluha odistí prepínaciu páku a nastaví ju do polohy, kedy je umožnený pohyb konštrukcie dynamického vertikalizátora do rozsahu 12° , čo zodpovedá maximálnemu možnému rozsahu.

3. Na počítači obsluha zapne Bluetooth, spustí hru Cube Dodger, ktorá sa nachádza na Pracovnej ploche a pripojí prototyp snímača polohy na 9V batériu. Ďalej obsluha spustí skript s názvom meno_hra.m v programe Matlab R2017.
4. Akonáhle sa v Matlabe objaví zobrazovanie aktuálnych hodnôt v okne Figure, Obsluha spustí hru tlačidlom START GAME, vyberie STANDARD mód a dobrovoľník hrá hru počas 2 minút.
5. Po uplynutí 2 minút obsluha, prípadne dobrovoľník, hru vypne, stopne spustený skript v programe Matlab, uloží Workspace a odpojí prototyp snímača polohy od napájania.
6. Obsluha uvoľní bedrovú podperu, dobrovoľník uvoľní plochu na státie a ohodnotí náročnosť a úroveň zábavy na stupnici od 1 do 5, pričom 1 znamená, že náročnosť bola príliš nízka a meranie nebolo vôbec zábavné a 5 znamená, že náročnosť bola veľmi vysoká a meranie bolo veľmi zábavné.
7. Meranie sa opakuje 8 až 10x.

Dátum a miesto merania: 19.12.2017, Brno

Protokol merania 2

ZAZNAMENANIE POHYBU SKUPINY DOBROVOĽNÍKOV ZA ÚČELOM ZÍSKANIA POLOHOVÝCH DÁT

Pomôcky a programy:

Dynamický vertikalizátor, prototyp snímača polohy, Matlab R2017a

Popis merania:

Meranie sa uskutoční na minimálne desiatich náhodne vybraných dobrovoľníkoch. Cieľom merania je získať polohové dáta z akcelerometra a gyroskopu, ktoré budú ďalej spracované a analyzované v programe Matlab R2017a. Proces získania dát zahŕňa plnenie úloh počas ktorých budú získavené dáta pomocou prototypu snímača polohy skonštruovaného vývojovou doskou Arduino Leonardo Pro Micro, akcelerometrom a gyroskopom MPU-6050 a Bluetooth modulom HC-05. Tento prototyp je umiestnený na nožičke dynamického vertikalizátora. Dáta budú do programu Matlab odosielané bezdrôtovo, budú uložené a pripravené na ďalšie spracovanie.

Postup merania:

1. Dobrovoľník sa postaví na plochu dynamického vertikalizátora určenú na státie. Obsluha zaistí kolieska, nastaví bedrovú podperu tak, aby tesne obopínala sedacie svaly dobrovoľníka, čím dobrovoľníka znehybní, prípadne nastaví kolenné podpery.
2. Obsluha odistí prepínaciu páku a nastaví ju do polohy, kedy je umožnený pohyb konštrukcie dynamického vertikalizátora do rozsahu 12° , čo zodpovedá maximálnemu možnému rozsahu.
3. Na počítači obsluha zapne Bluetooth a spustí skript s názvom meno.m v programe Matlab R2017.
4. Akonáhle sa v Matlabe objaví zobrazovanie aktuálnych hodnôt v okne Figure, obsluha vyzve dobrovoľníka, aby sa stojac na oboch nohách
 - naklonil čo najviac doprava a doľava,

- naklonil čo najviac vpred a vzad,
 - opísal štvorec pohybom tela,
 - vytvoril čo najrovnejšiu čiaru z pravej na ľavú stranu, prípadne naopak.
5. Ďalej obsluha vyzve dobrovoľníka, aby sa postavil na pravú nohu a zopakoval úlohy z bodu 4.
 6. Po splnení všetkých úloh obsluha stopne spustený skript v programe Matlab, uloží Workspace a odpojí prototyp snímača polohy od napájania.
 7. Obsluha uvoľní bedrovú podperu, dobrovoľník uvoľní plochu na státie a ohodnotí náročnosť a úroveň zábavy na stupnici od 1 do 5, pričom 1 znamená, že náročnosť bola príliš nízka a meranie nebolo vôbec zábavné a 5 znamená, že náročnosť bola veľmi vysoká a meranie bolo veľmi zábavné.
 8. Meranie sa opakuje 10 a viackrát.

Dátum a miesto merania: 17.3.2018, Brno

